

## PDF hosted at the Radboud Repository of the Radboud University Nijmegen

The following full text is a publisher's version.

For additional information about this publication click this link.

<http://hdl.handle.net/2066/99879>

Please be advised that this information was generated on 2021-03-03 and may be subject to change.

## What Does (and Doesn't) Make Analogical Problem Solving Easy? A Complexity-Theoretic Perspective

*Todd Wareham<sup>1</sup>, Patricia Evans<sup>2</sup>, and Iris van Rooij<sup>3</sup>*

Invited

### Abstract:

Solving new problems can be made easier if one can build on experiences with other problems one has already successfully solved. The ability to exploit earlier problem-solving experiences in solving new problems seems to require several cognitive sub-abilities. Minimally, one needs to be able to retrieve relevant knowledge of earlier solved problems and their solutions (solved-exemplar retrieval), to determine whether or not a retrieved problem is sufficient analogous to the problem at hand (analogy derivation), and to infer how the solution-method used for the old problem can be used for the new problem (candidate inference projection). All three processes have successfully been modeled under the framework of Structure-Mapping Theory (SMT). It has long been known that analogy derivation under SMT is computationally intractable, meaning that all (exact) algorithms implementing this ability run impractically long. In this paper we show that the same holds for the other two sub-processes. In sharp contrast to this theoretical intractability, empirical research reveals that in certain situations humans can quickly retrieve appropriate problem-exemplars and quickly make goal-relevant candidate inference projections. How can this speed of processing be explained within the framework of SMT? We consider several possible explanations, both existing and new, and assess their explanatory validity by performing computational-level complexity analyses. Our analyses not only reveal that explanations that have been conjectured to date are incomplete but also identify a set of complete explanations that can guide future empirical research on analogical problem solving.

### Keywords:

problem solving, analogy, Structure-Mapping Theory, computational complexity, intractability

---

<sup>1</sup>Memorial University of Newfoundland; <sup>2</sup> University of New Brunswick (Fredericton); <sup>3</sup>NICI, Radboud University Nijmegen

## 1. Introduction

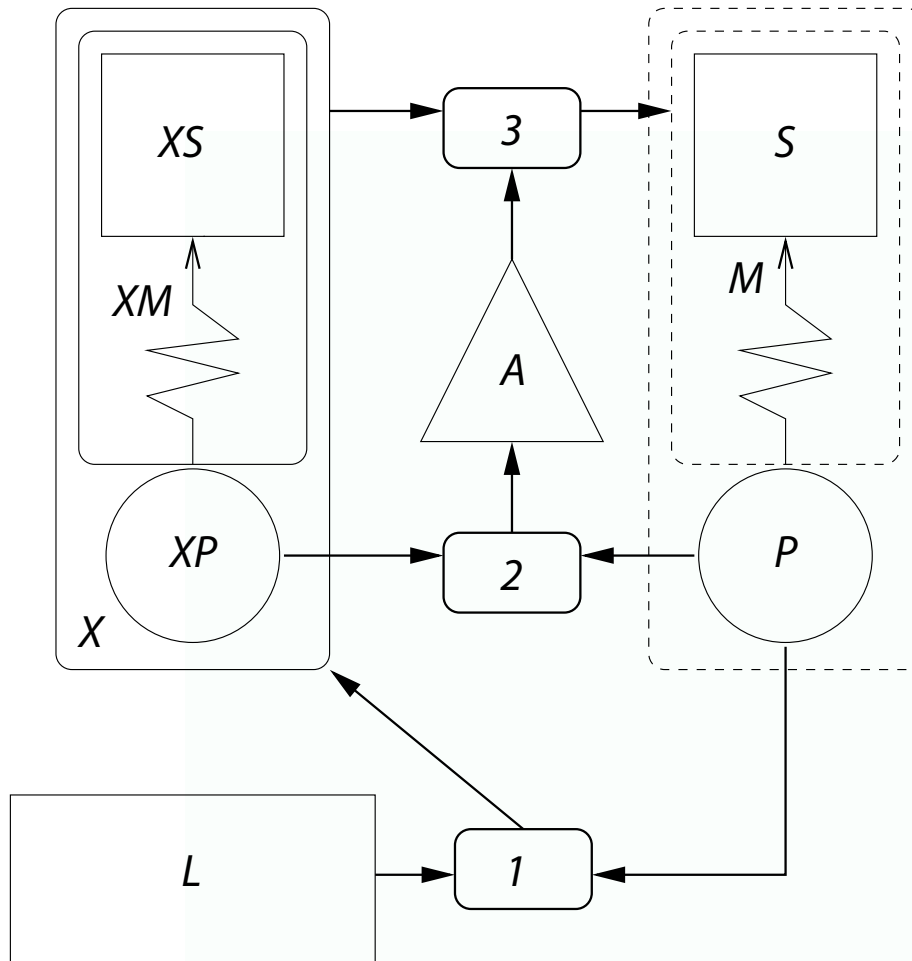
One technique for solving problems, especially if these problems are in a new and poorly understood domain, is to use analogies with instances of previously-solved problems to suggest solutions for new problems.<sup>1</sup> Sometimes, deriving the solution or solution-method from such an analogy involves the transfer of relatively little information from the solved-exemplar. This is the case in the classic problem of destroying a tumor with one or more beams of radiation (Duncker, 1945; Gick & Holyoak, 1980), where the required cumulative beam strength to destroy the tumor will also destroy intervening healthy tissue. When reminded of the previously solved problem of attacking a fortress reachable only by bridges that will collapse under the weight of the full required army (but which can be taken by several smaller armies simultaneously converging on the fortress via separate bridges), the analogy between the problem-instances (fortress  $\rightarrow$  tumor, size of army  $\rightarrow$  radiation dosage, bridge  $\rightarrow$  intervening healthy tissue) fairly directly suggests the solution (converging smaller armies  $\rightarrow$  converging lower-dosage beams). Other cases may require more transfer of information from the solved-exemplar. An example of this is solving a new mathematics or physics problem using similar solved problems (Falkenhainer, 1989; Novick & Holyoak, 1991); to solve the new problem, we not only need the initial analogy between similar entities and quantities in both the new and solved problem-instances, but we also need to transfer a possibly multistep chain of mathematical transformations that involve intermediate quantities not present in the original problem-instances and may require adaptation of some of these transformations to fit the new problem, *e.g.*, projection to the  $x$ - rather than the  $y$ -axis.

This suggests a general model of analogical problem solving (Gentner, 1989, pp. 215-217) shown in Figure 1. In this model, each solved problem consists of a problem-instance  $P$ , a solution  $S$ , and a solution-method  $M$  by which the solution is derived from the problem-instance. The processes invoked by this model are:

- **Solved-exemplar Retrieval:** Given a library  $L$  of solved-exemplars and a new problem-instance  $P$ , retrieve a solved-exemplar  $X = (XP, XM, XS)$  from  $L$  based on analogical similarity between the exemplar problem-instance  $XP$  and  $P$
- **Analogy Derivation:** Given  $X$  and  $P$ , derive an analogy  $A$  between  $XP$  and  $P$
- **Solution Adaptation:** Given  $X$ ,  $P$ , and  $A$ , use  $A$  to transfer and if necessary adapt

<sup>1</sup> There are several approaches to using such analogies to solve new problems. For example, analogies with previously solved problems can be used to guide the transfer of useful search-control strategies to new problems (van Lehn & Jones, 1991) or to build appropriate abstract mathematical models of a new domain for solving problems in that domain (Klenk & Forbus, 2009). In this paper, we will focus on the classical use of such analogies in problem-solving research, namely, in constructing either solutions or solution-methods for new problems from corresponding solutions or solution-methods in previously solved problems.

**Figure 1.** A General Model of Analogical Problem Solving (based on Figure 7.6 in Gentner, 1989). In this model, a new problem-instance  $P$  is screened against a library  $L$  of solved-exemplars to find the exemplar  $X$  whose problem-instance  $XP$  is most analogically similar to  $P$  (Box 1: Exemplar Retrieval). Once an appropriate analogy  $A$  between  $P$  and  $XP$  is derived (Box 2: Analogy Derivation), this analogy is used to construct either a solution-method  $M$  or a solution  $S$  for  $P$  from the solution-method  $XM$  or solution  $XS$  for  $X$  (Box 3: Solution Adaptation).



either  $XM$  or  $XS$  from  $X$  to create  $M$  or  $S$  for  $P$ .<sup>2</sup> Regardless of whether  $XM$  or  $XS$  is transferred, the initial part of this process involves extracting candidate inferences from those portions of  $XM$  and  $XS$  that are connected to the portion of  $XP$  mapped to  $P$  by  $A$  ((**Candidate Inference**)**Projection Derivation**).

<sup>2</sup> These two styles of analogical problem solving (construct solution or solution-method) are also known as problem solving by transformational analogy and derivational analogy, respectively (Carbonell, 1986; Schelhorn, Griego, & Schmid, 2006). As our analyses will apply to both, we do not focus on this distinction in this paper.

It is possible that inadequate results may be obtained at any phase above, *i.e.*, an irrelevant retrieved solved-exemplar, an insufficiently rich analogy, or a projection whose candidate inferences are false relative to the new problem-instance; hence, this model may be iterative, in that failure at one phase will cause backtracking and evaluation of other alternatives relative to an earlier phase.

Within the framework of Gentner's SMT of analogy derivation (Gentner, 1983), this model was first implemented in Falkenhainer's PHINEAS system (Falkenhainer, 1989). The individual processes in the model have subsequently been implemented using the software systems MAC/FAC (Forbus, Gentner, & Law, 1994) (solved-exemplar retrieval) and SME (Falkenhainer, Forbus, & Gentner, 1989; Forbus & Oblinger, 1990) (analogy/projection derivation). MAC/FAC and SME are both fast, and results produced by these systems are consistent with those produced by human subjects in various experiments *e.g.*, (Forbus, Gentner, & Law, 1994; Gentner et al., 2009; Gentner, Rattermann, & Forbus, 1993; Gentner et al., 1995). Yet, there are aspects of human performance in analogical problem solving that are not straightforwardly explained by these systems. In this paper we will focus on the following two target explanada:

1. The ability of experts to retrieve exemplars with good relationally-based (as opposed to merely surface-similarity-based) matches with new problem-instances (Blanchette & Dunbar, 2001; Novick, 1988) as opposed to the often abysmal retrieval abilities of people in general (Gick & Holyoak, 1983; Ross, 1989a; cf. Dunbar, 2001).
2. The general ability of people to quickly select those candidate inferences from the solved-exemplar's solution-method and solution that are most relevant to solving a new problem-instance (Spellman & Holyoak, 1996) (see also Gentner, 1989, pp. 215–217).

The challenge in explaining these abilities lies in the fact that the general computations underlying them seem to be computationally intractable when formalized under SMT-based models of analogical problem solving (we will substantiate this informal claim with formal proofs in our paper). How then can we explain that (expert) problem solvers are not paralyzed by such apparent intractability? Some informal conjectures have been made about what makes sub-processes in analogical processing tractable. For example, some researchers have conjectured that the speed of expert-level solved-exemplar retrieval is due to the facilitating effect of experts generally having small (abstracted) representations of solved-exemplars. Similarly, retrieving the most relevant candidate inference projection is thought to be fast because the derivation of arbitrary projections from analogies is known to be fast.

Although simulation research seems to support such conjectures (Gentner et al.,

2009), it does not yet constitute a proof that the postulated property (in this case, the relatively small size of representations, or the speed of deriving new inferences from analogies) is sufficient for making the computations tractable. In other words, as long as such explanations retain their conjecture status, they may well be incorrect at worst, and incomplete at best. Validation of said conjectures seems all the more important given that SMT-based models of analogical problem solving are increasingly finding application in real-world domains, such as education (Gentner et al., 2009; Ross, 1989b) and the design of AI-systems (Forbus & Hinrichs, 2006), and conjectures of tractability of sub-processes in these models are being used to make decisions and recommendations in those domains.

In this paper, we will show how computational complexity analysis—in particular, parameterized complexity analysis (Downey & Fellows, 1999)—can be used to formally validate conjectured explanations of tractability. We will first put forth computational-level characterizations of both relationally-based solved-exemplar retrieval and maximally-relevant projection derivation that can be abstracted from existing SMT-inspired algorithmic-level theories of these processes. We next use concepts and proof techniques from parameterized complexity theory to analyze these computational-level theories to determine what combinations of constraints can (and, perhaps more important, *cannot*) make these theories tractable. Our analyses not only reveal that existing conjectures of what makes sub-processes of analogical problem solving tractable under SMT-based models are incomplete, but also yield new explanations that can be formally proven to be complete. When combined with the results of the complexity analysis of analogy derivation under SMT in van Rooij et al. (2008), the results in this paper give a comprehensive overview of which constraints do and do not make analogical problem solving under SMT easy enough to be tractable.

### 1.1. Overview

The remainder of this paper is organized as follows. Section 2 gives an overview of analogical problem solving under SMT and our framework for using complexity analyses to investigate conjectures about what makes sub-processes of analogical problem solving under SMT tractable. Sections 3 and 4 present computational-level SMT models of the relationally-based solved-exemplar retrieval and maximally-relevant projection derivation sub-processes and our computational complexity results for each. In Section 5, we present a more general discussion in which we, among other things, compare our results with results previously obtained for a computational-level SMT model of the sub-process of analogy derivation. In Section 6, we conclude with take-home messages for theories of problem solving in particular, and theories of (domain-general) cognition in general. All mathematical definitions and proofs of results are given in the Appendix; this is to allow focus in the paper body on model formulations and the discussion of derived results.

## 2. Background

### 2.1. Analogical Processes under SMT

Since its proposal in 1983, SMT has evolved and broadened in scope. This has been motivated by both the results of psychological experiments and the act of implementing SMT in and then running simulations using various software systems, *e.g.*, SME (Falkenhainer, Forbus, & Gentner, 1989; Forbus & Oblinger, 1990), MAC/FAC (Forbus, Gentner & Law, 1994), and SEQL (Kuehne et al., 2000). That being said, there is a common “core” version of SMT dating from roughly 1989 as described in Falkenhainer, Forbus, and Gentner (1989). It is this version of SMT that we will use to formulate our computational-level models, and which will be described in this section.

Analogies are defined over conceptual structures. Conceptual structures are represented in SMT by predicate-structures composed of entities (*e.g.*, SUN, PLANET) and predicates relating those entities as well as other predicates (*e.g.*, ATTRACTS(SUN, PLANET), CAUSE(GRAVITY(MASS(SUN)), ATTRACTS(SUN, PLANET)). If the order of a predicate's arguments matter, *e.g.*, Greater(5, 3), the predicate is **ordered**; otherwise, it is **unordered**, *e.g.*, Equal( $x$ ,  $y$ ). A predicate-structure is ordered if all of its internal vertices are ordered; otherwise, if it contains at least one unordered predicate, it is unordered. Most predicate-structures contain both kinds of predicates; hence, in the remainder of this paper, unless otherwise stated, all predicate-structures will be unordered.

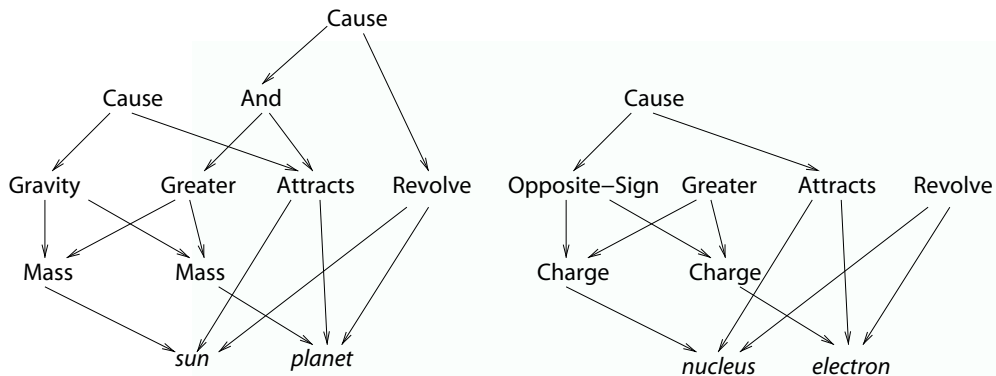
Predicate-structures are naturally represented as vertex-labeled directed acyclic-graphs in which entities are leaves, predicates are internal vertices, and predicates are linked to their arguments by arcs (which are themselves labeled, if the predicates are ordered) (Falkenhainer, Forbus, & Gentner, 1989, p. 12). Examples of two such graph-representations of predicate-structures corresponding to simplified models of the solar system and the Rutherford atom are shown in part (a) of Figure 2.

An **analogy** “ $T$  is (like) a  $B$ ,” where  $B$  and  $T$  are predicate-structures referred to as the **base** and **target**, respectively, is a mapping from a portion of  $B$  to a portion of  $T$  that satisfies the following three conditions (Gentner & Markman, 1997, p. 47):

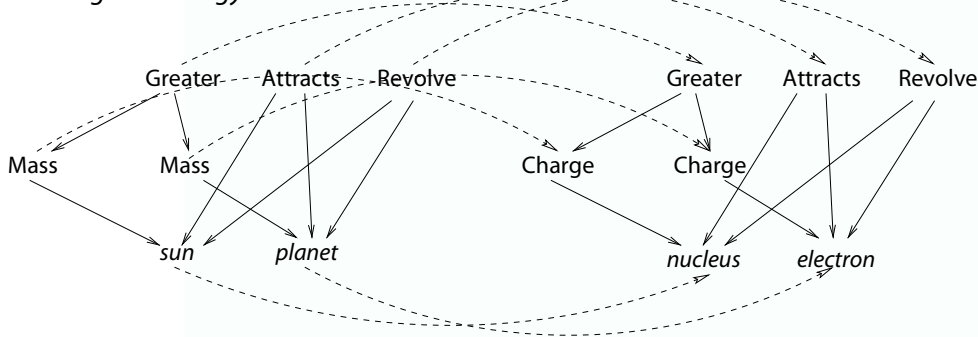
1. The mapping is **structurally consistent**, *i.e.*, it satisfies **parallel connectivity** (matching relations must have matching arguments) and **one-to-one correspondence** (any element in one predicate-structure matches at most element in the other predicate-structure).
2. **Relational focus**: The mapping must involve common predicates but need not involve common objects, *i.e.*, matched predicates must have the same name, argument, number, and order but matched objects need not have the same name.
3. **Systematicity**: The mapping tends to match interconnected, deeply nested predicate-substructures.

**Figure 2.** Analogical Mapping in Structure Mapping Theory. (a) Two graph representations of predicate-structures encoding simplified descriptions of the solar system (left) and the Rutherford model of the atom (right). (b) An analogy between the structures in (a), where dotted arrows indicate the mappings between corresponding pairs of predicates and objects. (c) A projection based on the analogy in (b) of candidate inferences from the richer Rutherford atom predicate-structure to the less rich solar system predicate-structure. Note that in this sub-figure, the solid polygons enclose the common analogical structure and the dashed polygons enclose the projection.

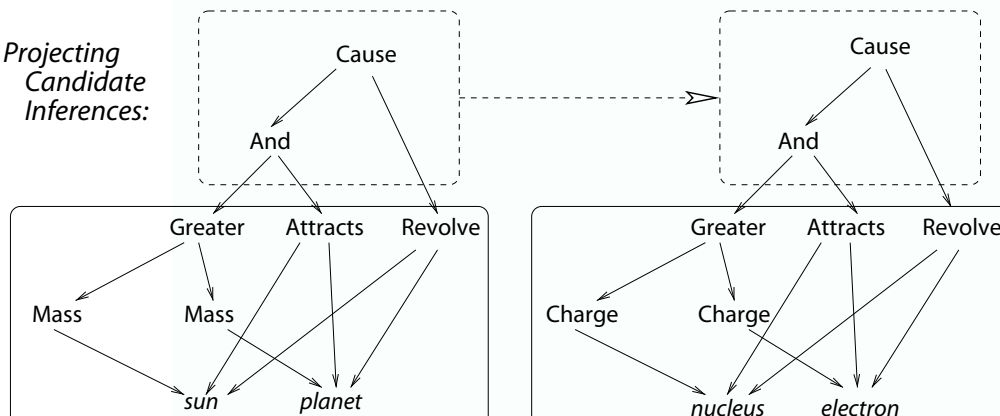
a) *Predicate Structures:*



b) *Deriving an Analogy:*



c) *Projecting Candidate Inferences:*





The most systematic analogy between a given base and target is also known as an **optimal analogy**. Such an analogy between two predicate-structures is shown in part (b) of Figure 2. Note that SMT does not require single-argument predicates that return numerical or enumerated values for the argument-object, *e.g.*, TEMPERATURE(SUN), CHARGE(ELECTRON), to match by name, *e.g.*, MASS(SUN) can match CHARGE(NUCLEUS) (Falkenhainer, Forbus, & Gentner, 1989).

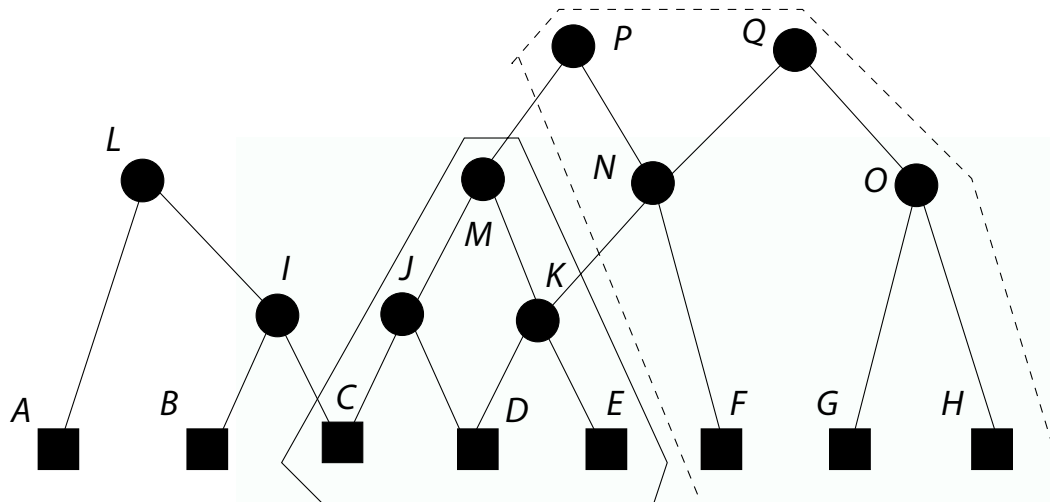
It is often the case (as in analogy-based problem solving [Falkenhainer, 1989] and metaphor interpretation [Bowdle & Gentner, 2005]) that one of the given conceptual structures is richer than the other; call this richer structure the **base** and the other the **target**. It has been shown that the transfer of structure from base to target is guided by an analogy (Clement & Gentner, 1991; Markman, 1997). In particular, the transferred structure consists of propositions called **candidate inferences** that are connected to predicates in the common analogical structure in the base but are not present in the target, and the transfer is called a **projection** (Falkenhainer, Forbus, & Gentner, 1989; Gentner & Markman, 1997). Such a projection is shown in part (c) of Figure 2.

Candidate inferences are so-called because they must be subjected to verification with respect to the target before they can be accepted as true and proper inferences. Each candidate inference consists of (i) a root predicate that is directly or indirectly connected to some predicate in the common analogical structure in the base and (ii) the collection of predicates and objects that are descendents of this root predicate in the base. Following Forbus et al. (1997), those descendents that are part of the common analogical structure are the inference's **support** and those that are outside are the inference's **extrapolation**. These concepts are all illustrated in Figure 3.

Various projections are shown in Figures 2, 3, and 4, where the common analogical structures are enclosed in solid polygons and the projections are enclosed in dashed polygons. In order to assure that each candidate inference in a projection is new and could not have been part of the analogy used to create it, we will assume that all projections are based on **maximal analogies**, which are analogies to which no additional pairs of predicates or objects in the base and target can be added. Note that all optimal analogies are maximal, but a maximal analogy is not necessarily optimal.

Derivation of analogies as described above has been implemented in the Structure-Mapping Engine (SME) (Falkenhainer, Forbus, & Gentner, 1989). As the original exhaustive-search version of SME could (depending on the given base and target) run in exponential time, SME is typically run in a much faster heuristic mode (Forbus & Oblinger, 1990) that is not guaranteed to find optimal analogies. As it can handle much larger and hence more realistic predicate-structures, heuristic-mode SME is a core component of many cognitive theories of analogy-based processes and their associated software implementations, including (as we will see in Section 3) the MAC/FAC system underlying the implementation of solved-exemplar retrieval.

**Figure 3.** Concepts in Candidate Inference Projection. This figure shows a base predicate-structure with a common analogical structure (enclosed in a solid polygon) relative to some target predicate-structure (not shown) and the maximum-size projection for that common structure (enclosed in a dashed polygon) and the characteristics of each candidate inference in this projection. Note that  $\{B, C, I\}$  and  $\{A, B, C, I, L\}$  are not valid candidate inferences, as predicates  $I$  and  $L$  only connect to the common analogical structure through an object ( $C$ ).



Candidate Inference	Root Predicate	Path To Base Analogy	Inference Support	Inference Extrapolation
$\{D, E, F, K, N\}$	N	$N \rightarrow K$	$\{D, E, K\}$	$\{F, N\}$
$\{C, D, E, F, J, K, M, N, P\}$	P	$P \rightarrow M$	$\{C, D, E, J, K, M\}$	$\{F, N, P\}$
$\{D, E, F, G, H, K, N, O, Q\}$	Q	$Q \rightarrow N \rightarrow K$	$\{D, E, K\}$	$\{F, G, H, N, O, Q\}$

## 2.2. A Methodology for Validating Explanations of Tractability

Recall that our aim in this paper is to assess the validity of conjectured explanations of the tractability of certain sub-processes of analogical problem solving. As we are interested in the (in)tractability inherent in a sub-process theory itself, and not in particular (potentially inefficient) algorithmic implementations of that theory, we will focus our analysis on Marr's computational level (Marr, 1981). That is, our analyses will pertain to the input-output mappings postulated to explain the target sub-processes.

To make clear what type of validation we are after, we first explain what we understand as a failure to explain the speed of some cognitive process. A computational-level theory  $\psi: I \rightarrow O$  is said to fail to explain the speed (minimally, the tractability) of a given (or any) cognitive (sub)process if there exists no efficient algorithm for tractably computing the

function  $\psi$ . For example, algorithms whose running time is super-polynomial in the size of the input representations are generally inefficient for all but small input sizes. Such algorithms require an amount of time that cannot be upperbounded by any polynomial function  $n^c$  (where  $n$  is a measure of the input size and  $c$  is some constant). Examples are exponential-time algorithms, which require time that can (at best) be upperbounded by some exponential function  $c^n$ . To see that, for instance, exponential-time algorithms are impractical for even medium-sized inputs, consider that  $2^n$  is already more than the number of seconds in a year for  $n = 25$ .

As we will show, SMT-based computational-level theories of relationally-based solved-exemplar retrieval and maximally-relevant projection derivation have the property that all algorithms for computing them are super-polynomial (*i.e.*, exponential or worse) in the input size (for proofs refer to Sections 3 and 4). Note that in analogical problem solving, input size cannot generally be assumed to be small, as our knowledge of different problems can be quite rich and encoding such knowledge in a predicate-structure may lead to quite large representations. This means that these SMT-based computational-level theories of solved-exemplar retrieval and maximally-relevant projection derivation cannot by themselves explain the tractability of these processes; in fact, the very formulation of these theories seems to contradict the tractability of the modeled processes.

The above does not yet mean that the tractability of the modeled processes cannot be explained by enriching the theories by some theoretical constraints. To identify such constraints, we will use techniques to prove fixed-parameter tractability of a function  $\psi$  for one or more *parameters* of inputs in  $I$  (see also Downey and Fellows, 1999).

**Definition 1 Fixed-parameter (fp-) tractability.** Let  $\psi : I \rightarrow O$  be an input-output function with input parameters  $k_1, k_2, \dots, k_m$ . Then  $\psi$  is said to be fixed-parameter tractable for parameter-set  $K = \{k_1, k_2, \dots, k_m\}$  if there exists at least one algorithm that computes  $\psi$  for any input of size  $n$  in time  $f(k_1, k_2, \dots, k_m)n^c$ , where  $f(\cdot)$  is an arbitrary computable function and  $c$  is a constant. If no such algorithm exists then  $\psi$  is said to be fixed-parameter intractable for parameter-set  $K$ .

In other words, a function is fp-tractable for a parameter-set  $K$  if all super-polynomial-time complexity inherent in computing the function can be confined to the parameters in  $K$ . It then follows from the definition of fp-tractability that if an intractable function  $\psi$  is fp-tractable for parameter-set  $K$ , then  $\psi$  can be efficiently computed even for large inputs, provided only that all the parameters in  $K$  are small. This means that if  $\psi$  is postulated as an explanation of the functional form of the input-output mapping computed by a given process, then the speed of that process in certain situations can be explained by postulating that the parameters in  $K$  are small exactly in those situations.

Note that in this way of conceptualizing "explanation," the functional form and the

speed of a cognitive process are treated as two distinct and separable target explananda. Alternatively, one could claim that it is really the enriched theory  $\psi' : I' \rightarrow O$ , where  $I' \subset I$  denotes the input domain constrained to inputs have small parameter values for all parameters in  $K$ , that is explaining both the functional form and the speed of the cognitive process under study. The latter perspective would be more consistent with the model-revision approach to tractable explanation advocated by van Rooij (2008) and van Rooij and Wareham (2008). Although we believe the latter is conceptually cleaner, in the context of this paper we will take the former conceptualization of explanation because we are interested particularly in explanations of speed/tractability relative to a given theory of the functional form of a cognitive process.

Lastly, we note that the above allows us to distinguish between complete and incomplete explanations of the tractability of a function  $\psi$ .

**Observation 1** *There exist functions  $\psi$ , and parameter-sets  $K'$ ,  $K$ , such that  $K' \subset K$  and  $\psi$  is fp-tractable for  $K$  but not for  $K'$ .*

The observation implies that there are input constraints (e.g., all parameters in  $K'$  are small) that by themselves cannot explain the tractability of a cognitive process modeled by a function  $\psi$ , yet are part of such an explanation. In such cases we would say the assumption that all parameters  $K'$  are small for the relevant situation is an incomplete explanation of the tractability of the process, whereas replacing  $K'$  by  $K$  yields a complete explanation.

Before closing this section, we wish to clarify one point. Although our formal analyses will work under the assumption that the analyzed models predict the outputs of human computations *exactly*, we conjecture they also apply under the assumption that the analyzed models predict the outputs of human computations *approximately*, provided that by “approximately” one means that the approximate prediction does not differ from the exact prediction by more than some small constant factor. This conjecture is based on the observation that many NP-hard functions cannot be efficiently approximated (Arora, 1998; Ausiello et al., 1999), and almost all are inapproximable if only a constant sized error is allowed (Garey & Johnson, 1979; van Rooij, Wright, & Wareham, to appear). We realize that this conjecture may be counterintuitive to some cognitive scientists, as it is frequently suggested in the cognitive science literature that computationally intractable functions can (often or always) be approximately computed efficiently (e.g., Love, 2000; Chater et al., 2003; Chater, Tenenbaum, & Yuille, 2006). Popular though this belief is, it does not accord with the state of the art in computational complexity theory (cf. Kwisthout, Wareham, & van Rooij, 2010). For this reason, in our analyses we will not treat approximate computations explicitly. For ideas on how approximation can be explicitly taken into account in fixed-parameter tractability analyses, we refer the reader to Hamilton et al. (2007) and Marx (2008).

### 3. Relationally-based Solved-Exemplar Retrieval

When engaging in analogical problem solving, retrieving appropriate exemplars of earlier solved problems can be a challenging task. Experimental evidence suggests that though humans prefer systematic relation-rich matches between new problem-instances and solved-exemplars as being useful for problem solving, in practice, people typically retrieve exemplars that are based on surface similarities between the entities involved (**literal similarity**), even though these matches are often less useful or even misleading (Gentner, Rattermann, & Forbus, 1993; Gick & Holyoak, 1983; Ross, 1989a; cf. Dunbar, 2001). That being said, relationally-based exemplar retrievals do occur, and are much more frequently done by experts (Blanchette & Dunbar, 2001; Novick, 1988) or people in general who are able to compare problem-instances prior to retrieval (Gentner et al., 2009). These improvements in performance have been attributed to common (re)coding of exemplars and probes (Medin & Ross, 1989) and the use of small schema abstractions as both stored exemplars and probes (Gentner et al., 2009).

Within the framework of SMT, solved-exemplar retrieval is implemented in the MAC/FAC system (Forbus, Gentner, & Law, 1994). MAC/FAC was explicitly designed to duplicate the observed human behaviors described above. It uses a two-phase search process, in which a computationally cheap matching process based on feature-vector similarity screens the whole library of exemplars against a specified probe (MAC: "Many Are Called"), and a small subset of those exemplars with highest feature-vector similarity to the probe is subsequently assessed in parallel against the probe by a computationally expensive analogy-based matcher (FAC: "Few Are Chosen"). As all exemplars with good analogies with the probe may not be appropriate for solving the problem-instance encoded in the probe, the FAC phase returns up to a small number (typically three) of these exemplars. In the interests of computational psychological plausibility, MAC/FAC uses the fast greedy heuristic version of SME to implement FAC.

Comparisons of MAC/FAC simulations with the results of human experiments have shown that MAC/FAC broadly mimics the way human beings typically retrieve exemplars (Forbus, Gentner, & Law, 1994; Gentner et al., 1993); moreover, MAC/FAC simulations when using schema abstractions as probes and exemplars show similar improvements in retrieval performance to those of human beings (Gentner et al., 2009). However, on the basis of simulations alone (which do not operate on all possible inputs), it is impossible to show for certain that a system like MAC/FAC will always retrieve relationally-based exemplars in a psychologically plausible amount of time relative to a particular type of input like schema abstractions. Definitive evaluations of such explanations require the analysis of a computational-level theory of relationally-based solved-exemplar retrieval, as will be done in the remainder of this section.

### 3.1. Formulating the Computational-level Theory

The first decision that must be made when formulating a theory of relationally-based solved-exemplar retrieval is to decide just how relationally-based these retrieved exemplars should be. This has not been explicitly discussed in the literature; however, as analogy derivation under SMT prefers the most systematic analogies, solved-exemplar retrieval can similarly be construed to prefer those exemplars that have the most systematic analogies with the given new problem-instance. Hence, for simplicity, we will consider the retrieval of those solved-exemplars from a library  $L$  whose optimal analogies with the new problem-instance probe  $P$  have the largest possible systematicity over all optimal analogies of  $P$  with the members of  $L$ .

As there may be several such exemplars for a given  $L$  and  $P$ , not all of which may be appropriate for solving the new problem-instance, we would like to see as many of them as possible; however, in the interests of efficiency (and following the operation of MAC/FAC), we may want to bound the number returned at any one time. Both of these requirements can be accommodated by retrieving an arbitrary subset of these optimal-analogy solved-exemplars that is the largest possible but also of size at most some specified bound  $b$ . For example, if  $b = 3$  and there are only two exemplars in  $L$  that have the best possible optimal analogies over  $L$  with  $P$ , then both of those exemplars are returned; however, if there are five such exemplars, an arbitrary three of those five will be returned.

The above gives the following computational-level theory:

EXEMPLAR RETRIEVAL (ER)

*Input:* A library  $L$  of predicate-structures, a probe predicate-structure  $P$ , and an integer  $b$ .

*Output:* A largest possible subset  $L' \subseteq L$  of size at most  $b$  such that each member of  $L'$  has an optimal analogy with  $P$  of the largest possible systematicity over all optimal analogies between  $P$  and members of  $L$ .

The formal version of this input-output mapping, including details of all entities therein, is given in Appendix Section A.4. Note that this mapping returns those exemplars in  $L$  that have optimal analogies with  $P$  without specifying what these analogies are—hence, it is ideal for investigating the process of solved-exemplar retrieval in isolation (as opposed to the conflation of retrieval and analogy derivation in MAC/FAC).

### 3.2. Results and Discussion

We start by stating our main intractability result for the computational-level theory EXEMPLAR RETRIEVAL as defined in Section 3.1:

**Table 1.** Overview of parameters considered in our fixed-parameter tractability analyses of EXEMPLAR RETRIEVAL.

Name	Definition
$ B $	Size of largest solved-exemplar in $L$
$ T $	Size of $P$
$p$	Maximum number of predicates in $L$ and $P$
$o$	Maximum number of objects in $L$ and $P$
$h$	Maximum order of predicates in $L$ and $P$
$a$	Maximum number of arguments per predicate in $L$ and $P$
$f$	Maximum frequency of a given predicate in $L$ and $P$
$t$	Maximum number of predicate-types in $L$ and $P$
$ L $	Number of solved-exemplars in library $L$
$b$	Maximum number of returned solved-exemplars

**ER1.** EXEMPLAR RETRIEVAL *is intractable, in the sense that all algorithms computing this input-output function require super-polynomial time.*<sup>3</sup>

This result establishes that there are no polynomial-time algorithms (including MAC/FAC) that can retrieve optimal relationally-based exemplars over all inputs. It also means that some strong constraints must be assumed to apply to the input domain of EXEMPLAR RETRIEVAL for the theory to be able to explain speedy retrieval of relationally-based solved-exemplars by human experts. To investigate which constraints suffice for rendering EXEMPLAR RETRIEVAL tractable we performed fp-tractability analyses relative to the parameters in Table 1. This table includes parameters relevant for testing the validity of existing explanation of tractability of relationally-based solved-exemplar retrieval under SMT (modeled by EXEMPLAR RETRIEVAL) as well as some new explanations that we conjecture ourselves.

Our analyses return the following set of fp-intractability results:

**ER2.** EXEMPLAR RETRIEVAL *is fp-intractable for parameter-set*  $\{|B|, h, a, t, |L|, b\}$ .

**ER3.** EXEMPLAR RETRIEVAL *is fp-intractable for parameter-set*  $\{|T|, h, a, t, |L|, b\}$ .

**ER4.** EXEMPLAR RETRIEVAL *is fp-intractable for parameter-set*  $\{h, a, f, b\}$ .

These results are sobering—they show that constraining many of the considered parameters to take relatively small values, whether individually or combined, does not help to

<sup>3</sup> The claim follows from our NP-hardness proof for EXEMPLAR RETRIEVAL (Appendix Section A.4, Lemmas 4 and 5), plus the widely believed mathematical conjecture that  $P \neq NP$ . In this paper we, like SMT theorists (see, e.g., Markman and Gentner, 2000, p. 507), will not question the truth of this conjecture (see Fortnow, 2009; Garey & Johnson, 1979 for reasons why not to).

render relationally-based exemplar retrieval under SMT tractable (this holds true even when many of these parameters are restricted to very small constant values; see Appendix Section A.4). Of particular interest are Results ER2 and ER3, which suggest that the explanation by Gentner et al. (2009) of fast relationally-based exemplar retrieval purely in terms of either exemplars or probes as small schema abstractions ( $|B|, |T|$ ) is untenable, even when the size of the library  $L$  and the maximum number of returned exemplars are trivial (i.e., in our proofs,  $|L| = b = 2$ ; see Appendix Section A.4).

This is not to say that no viable explanations of tractability based on schema abstractions exist; rather, it says that such explanations are as of yet incomplete. Tractability may be achieved when additional constraints are introduced, as confirmed by the following set of fp-tractability results:

**ER5.** EXEMPLAR RETRIEVAL *is fp-tractable for parameter-set*  $\{o\}$ .

**ER6.** EXEMPLAR RETRIEVAL *is fp-tractable for parameter-set*  $\{p\}$ .

**ER7.** EXEMPLAR RETRIEVAL *is fp-tractable for parameter-set*  $\{f, t\}$ .

**ER8.** EXEMPLAR RETRIEVAL *is fp-tractable for the parameter-sets*  $\{|B|, f\}$  and  $\{|T|, f\}$ .

Result ER8 is of particular interest, as it resurrects the schema-abstraction-based explanation, albeit with an additional and unanticipated restriction (viz., that additionally  $f$  is small for such representations). It seems reasonable to conjecture that  $f$  can be small even for large predicate-structures and this can be verified both by experiment and by examining the values of  $f$  in predicate-structures used in SMT-based simulation studies. If we assume that the new problem-instances are also small (which may be the case, especially if the new problem-instance is incompletely or partially understood), Results ER5 and ER6 allow additional abstraction-based explanations (in that both  $|B|$  and  $|T|$ , and not just  $|B|$  or  $|T|$ , are small when  $o$  and  $p$  are small); again, such an assumption would require empirical verification.

#### 4. Maximally-relevant projection derivation

When an analogy has been drawn between two predicate-structures, there are typically so many candidate inferences that may be projected between these structures that it is psychologically implausible to assume that all of these inferences are projected. Hence, it is typically assumed that projected candidate inferences are selected relative to some criterion, e.g., goal relevance (Spellman & Holyoak, 1996), adaptability (Keane, 1996), and support and/or extrapolation (Forbus et al., 1997; see also Gentner 2002, p. 107).

As we are interested here in analogical problem solving, we will focus on the goal relevance criterion, i.e., derive the projection from  $B$  to  $T$  that has the most candidate



inferences relative to set  $S$ , where  $S$  is a subset of the predicates in  $B$  that are relevant to solving the problem-instance encoded in  $B$ . We will call any such projection a **maximally-relevant projection**. For example, when projecting the solution and solution-method from a solved-exemplar to a new instance of a mathematics or physics problem, projections that include individual predicates encoding key parts of the solution or solution-method, *e.g.*, YIELDS, PROVES, IMPLIES, CAUSES, would be useful, and those projections that incorporate as many such predicates as possible would be most useful (and hence most relevant) to the goal of solving the new problem-instance.

Within the framework of SMT, projection derivation is implemented in the SME system (Falkenhainer, Forbus, & Gentner, 1989). SME uses a polynomial-time process that constructs the maximum-size projection from a given analogy by “growing” that projection upward from the common analogical structure in the base (Falkenhainer, Forbus, & Gentner, 1989). However, neither this algorithm nor the only published algorithm for creating relevant projections relative to a specified  $S$  (Forbus & Oblinger, 1990) (which also runs in polynomial time) has been proven to produce maximally-relevant projections relative to any set of inputs, let alone those occurring in human analogical problem solving. Hence, definitive evaluations of claims that algorithms like those in Falkenhainer, Forbus, and Gentner (1989) or Forbus and Oblinger (1990) can explain fast maximally-relevant projection derivation require the analysis of a computational-level theory of maximally-relevant projection derivation, as will be done in the remainder of this section.

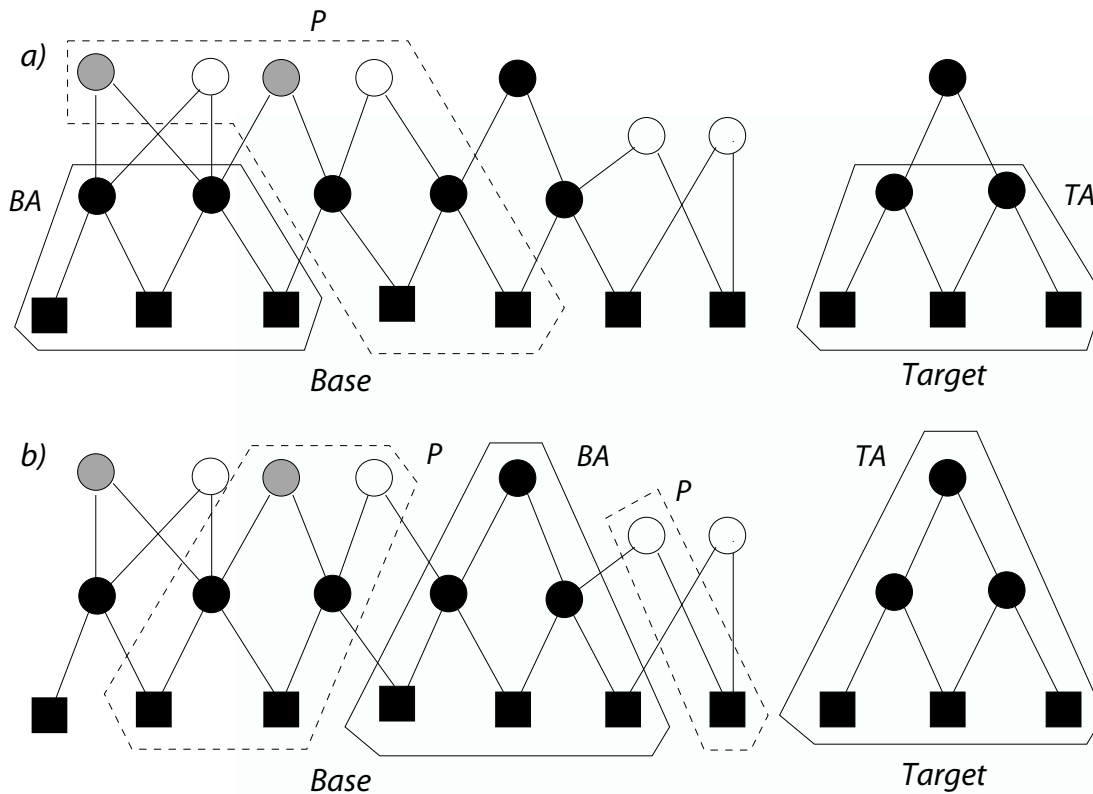
#### **4.1. Formulating the Computational-level Theory**

At first glance, it would seem that any computational-level theory of maximally-relevant projection derivation must be based on the accepted SMT theory of projection derivation described in Section 2.1. Recall that this theory assumes that a projection is both created after the derivation of and based on an optimal analogy between the given base  $B$  and target  $T$ . Both of these assumptions are (and hence the theory is) consistent with experimental observations (Clement & Gentner, 1991; Markman, 1997; see also Gentner & Colhoun, 2010, pp. 37-38). However, in the context of maximally-relevant projection derivation, this theory is inadequate: not only has it been known for more than 20 years that all optimal analogies do not necessarily yield maximally-relevant projections (Falkenhainer, 1989), but, as shown in Figure 4, maximally-relevant projections need not even be associated with optimal analogies.

These problems can be resolved in an elegant and simple manner by making the following assumptions:

1. Given that the maximally-relevant projection in our example in Figure 4 is based on a maximal analogy, assume that maximally-relevant projections are based on maximal rather than optimal analogies.

**Figure 4.** Optimal Analogies Need Not Give Maximally-Relevant Projections. (a) A maximal analogy between given base and target predicate-structures. (b) An optimal analogy between the base and target structures. Relevant predicates in set  $S$  are colored gray. Each analogy has associated base and target common structures  $BA$  and  $TA$  and the maximally-relevant projection relative that analogy  $P$  (which may, as in part b, be comprised of two disjoint sub-projections). Note that in this figure, the solid polygons enclose the common analogical structures and the dashed polygons enclose the projections.



2. Given that all maximal analogies do not yield maximally-relevant projections (as the optimal analogy in part b of Figure 4 is itself maximal), assume that maximally-relevant projections are derived *at the same time as* (rather than after) their underlying (maximal) analogies.

Though these assumptions may initially seem to contradict SMT and experimental observations, they do not—indeed, they are actually consistent with both. That optimal analogies are preferable in some contexts does not rule out that maximal analogies are preferable in the context of analogical problem solving (and more specifically, the derivation of

maximally-relevant projections), and the only experiments suggesting that projections are derived after analogies (Clement & Gentner, 1991; Markman, 1997; Gentner & Colhoun, 2010) were not carried out in the context of analogical problem solving, where projections and analogies may indeed be derived together. Given the above, we will thus base our computational-level theory of maximally-relevant projection derivation on these assumptions.

Further constraints on the forms of the projections themselves are suggested by the following two modes for projection derivation:

1. Derive the projection with the largest number of predicates from the relevant-predicate set  $S$ .
2. Derive a projection with the largest number of relevant predicates from  $S$  that are in a "shell" enclosing the common analogical structure in the base, where this shell is specified by a maximum distance  $d$  of a predicate from that common structure, *i.e.*, the maximum number  $d$  of edges on the shortest path from a predicate to that common analogical structure (see Figure 3).

The former is more straightforward. However, the latter allows an iterative strategy in which predicates that are closer to the common analogical structure in the base are derived before predicates that are farther away. In cases where there is reason to believe that some candidate inferences in the maximally-relevant projection may be false when projected to the new problem-instance, this iterative approach may save both computation and verification time.

The above gives the following computational-level theory:

PROJECTION DERIVATION

*Input:* Two predicate-structures  $B$  and  $T$ , a subset  $S$  of the predicates in  $B$ , and integers  $k$  and  $d$ .

*Output:* A maximal analogy  $A$  between  $B$  and  $T$  and a projection  $P$  relative to  $A$  and  $B$  such that  $A$  and  $P$  include at least  $k$  predicates from  $S$  and each such predicate has distance at most  $d$  from the common structure of  $A$  in  $B$ .

The formal version of this input-output mapping, including details of all entities therein, is given in Appendix Section A.5. Note that for appropriate values of  $k$ ,  $d$ , and  $S$ , the requested analogy/projection pair may not exist for the given  $B$  and  $T$  (as signaled by the special output  $\perp$ ). In these situations, it is assumed that the expectations for a projection relative to  $B$ ,  $T$ , and  $S$  will be modified in some fashion (by changing the values of  $k$  and/or  $d$  and/or the members of  $S$ ) until a satisfactory projection is produced.

**Table 2.** Overview of parameters considered in our fixed-parameter tractability analyses of PROJECTION DERIVATION.

Name	Definition
$ B $	Size of $B$
$ T $	Size of $T$
$p$	Maximum number of predicates in $B$ and $T$
$o$	Maximum number of objects in $B$ and $T$
$h$	Maximum order of predicates in $B$ and $T$
$a$	Maximum number of arguments per predicate in $B$ and $T$
$f$	Maximum frequency of a given predicate in $B$ and $T$
$t$	Maximum number of predicate-types in $B$ and $T$
$ S $	Size of predicate-set $S$
$k$	Minimum number of members of $S$ in analogy and projection
$d$	Maximum distance of predicates/objects from base analogy in projection

#### 4.2. Complexity Results and Discussion

We start by stating our main intractability result for the computational-level theory projection derivation as defined in Section 4.1:

**PD1.** PROJECTION DERIVATION *is intractable, in the sense that all algorithms computing this input-output function require super-polynomial time.*<sup>4</sup>

Note that this intractability is not due to the complexity of deriving the analogy itself, as maximal analogies can be derived in polynomial time.<sup>5</sup> Result PD1 establishes that there are no polynomial-time algorithms (including the heuristic described in Forbus & Oblinger, 1990) that can compute maximally-relevant projections over all inputs. It also means that some strong constraints must be assumed to apply to the input domain of PROJECTION DERIVATION for the theory to be able to explain speedy computation of maximally-relevant projections by human beings. To investigate which constraints suffice for rendering PROJECTION DERIVATION tractable, we performed fp-tractability analyses relative to the parameters in Table 2. This table includes parameters relevant for testing the validity of explanations of tractability of maximally-relevant projection derivation under SMT (modeled by PROJECTION DERIVATION) analogous to those proposed in the literature for EXEMPLAR RETRIEVAL as well as some new explanations that we conjecture ourselves.

<sup>4</sup> The claim follows from our NP-hardness proof for PROJECTION DERIVATION (Appendix Section A.5, Lemmas 10 and 11). Again we assume  $P \neq NP$ ; see also Footnote 1.

<sup>5</sup> A simple algorithm for this is as follows: select an arbitrary mapping between collections of objects in  $B$  and  $T$  of the same size, and the analogy “grown” upward from this object mapping by the algorithm underlying the proof of Claim 5 in van Rooij et al. (2008) will be maximal.

Our analyses return the following set of fp-intractability results:

**PD2.** PROJECTION DERIVATION is *fp-intractable* for parameter-set  $\{|B|, h, a, t, |S|, k, d\}$ .

**PD3.** PROJECTION DERIVATION is *fp-intractable* for parameter-set  $\{|T|, h, a, t, k, d\}$ .

**PD4.** PROJECTION DERIVATION is *fp-intractable* for parameter-set  $\{h, a, f, d\}$ .

These results show that constraining many of the considered parameters to take relatively small values, whether individually or combined, does not help to render maximally-relevant projection derivation under SMT tractable (this holds true even when many of these parameters are restricted to very small constant values; see Appendix Section A.5). Of particular interest is Result PD3, which suggests that explaining fast maximally-relevant projection derivation purely in terms of small new problem-instances ( $|T|$ ) (which would be possible if such instances are represented as small schema abstractions (Gentner et al., 2009; see also Section 3.2) is untenable.

This is not to say that no viable explanations of tractability exist. Tractability may be achieved when different constraints are considered, as confirmed by the following set of fp-tractability results:

**PD5.** Projection derivation is *fp-tractable* for parameter-set  $\{o, |S|\}$ .

**PD6.** Projection derivation is *fp-tractable* for parameter-set  $\{p\}$ .

**PD7.** Projection derivation is *fp-tractable* for parameter-set  $\{f, t\}$ .

**PD8.** Projection derivation is *fp-tractable* for parameter-sets  $\{|B|, f\}$  and  $\{|T|, f\}$ .

As the bases in instances of maximally-relevant projection derivation will be fully solved-exemplars that include solution-methods and solutions, it is unlikely that  $|B|$  will be small, and thus Results PD5, PD6, and PD7 and the first parameter-set in Result PD8 are of limited interest. However, if new problem-instances and  $f$  are small, as conjectured in Section 3.1, the second parameter-set in Result PD8 is promising. Hence, though PROJECTION DERIVATION is not tractable in general, there are constraints that make it so.

A final caveat is in order: the utility of the results and discussion above is predicated on the correctness of our theory of maximally-relevant projection derivation (or, more specifically, the assumptions underlying this theory). This needs to be assessed by empirical research. However, regardless of whether or not our theory is correct, any correct theory must address the problems with the currently accepted SMT-based theory pointed out in Section 4.1.

The existence of such problems nicely illustrates the more general lesson that, when modeling a cognitive process, it is important to consider both the different contexts in which that process is called and other processes with which that process interacts. In

retrospect, even though SMT from its inception saw projection derivation as important, assumptions about the nature of projection and analogy derivation in general were made without considering the roles of either in other processes such as solution adaptation. Given that analogy is conjectured to be central to many cognitive processes (Gentner, 2003, 2010; Gentner & Colhoun, 2010), it is critical to model and analyze these other analogy-based processes to see how valid such assumptions are in a wider cognitive context.

## 5. General Discussion

Having previously discussed specific implications of our results for relationally-based solved-exemplar retrieval and maximally-relevant projection derivation in Sections 3.2 and 4.2, in this section, we would like to discuss several more general implications of these results. Specifically, we will put our results in a broader context by comparing them with complexity results for analogy derivation presented in van Rooij et al. (2008) and Appendix Section A.3 and then with the model of analogical problem solving described in Section 1.

The analogy derivation phase in the model of analogical problem solving described in Section 1 has the following computational-level theory under SMT:

ANALOGY DERIVATION

*Input:* Two predicate-structures  $B$  and  $T$ .

*Output:* An analogy  $A$  between  $B$  and  $T$  of optimal systematicity.

Consider the following summary of known complexity results for analogy derivation under SMT as rephrased in the notation used in this paper:

**AD1.** (van Rooij et al., 2008, Corollary 2; Veale and Keane, 1997) ANALOGY DERIVATION is polynomial-time intractable.

**AD2.** (van Rooij et al., 2008, Claims 4 and 6) ANALOGY DERIVATION is fp-intractable for parameter-sets  $\{|B|, h, a, t\}$  and  $\{|T|, h, a, t\}$ .<sup>6</sup>

**AD3.** (van Rooij et al., 2008, Claim 1) ANALOGY DERIVATION is fp-intractable for parameter-set  $\{h, a, f, t\}$ .

**AD4.** (van Rooij et al., 2008, Claim 5) ANALOGY DERIVATION is fp-tractable for parameter-set  $\{o\}$ .

<sup>6</sup> Note that Claim 4 as written in van Rooij et al. (2008) states that ANALOGY DERIVATION is fp-intractable for parameter-set  $\{|B|, h, a, t, p\}$ , which seems to contradict Result AD5. However, the addition of  $p$  to this parameter-set was a typo in van Rooij et al. (2008), meaning that Result AD5 is correct.

**AD5.** (Appendix Section A.3) ANALOGY DERIVATION is fp-tractable for parameter-set  $\{p\}$ .

**AD6.** (Appendix Section A.3) ANALOGY DERIVATION is fp-tractable for parameter-set  $\{f, t\}$ .

**AD7.** (Appendix Section A.3) ANALOGY DERIVATION is fp-tractable for parameter-sets  $\{|B|, f\}$ , and  $\{|T|, f\}$ .

From the summary above, it is evident that (modulo the deletion of parameters specific to individual theories), EXEMPLAR RETRIEVAL, ANALOGY DERIVATION, and PROJECTION DERIVATION share the same fp-tractability and intractability results. This may initially seem odd, as neither relationally-based solved-exemplar retrieval nor maximally-relevant projection derivation is explicitly deriving or returning optimal analogies. However, on closer examination of the proofs underlying these results, this is not so surprising: all three theories invoke a bounded analogy evaluation computation, *i.e.*, is there an analogy between  $B$  and  $T$  of systematicity  $\geq k?$ , which is actually the computation relative to which all intractability results in van Rooij et al. (2008) are proven, and all three theories can be implemented using algorithms that can examine all possible maximal analogies between a given  $B$  and  $T$  (which the fp-algorithms given here all do).

As these mechanisms seem rather generally applicable, we conjecture that many (if not all) analogy-based processes will share the same roster of fixed-parameter results relative to the parameters considered here. It seems likely that this unity may shatter and the various theories of analogy-based processes will vary in terms of their fp-tractabilities and intractabilities when other shared parameters are considered (indeed, the fact that PROJECTION DERIVATION is not currently known to be fp-tractable for parameter-set  $\{o\}$  may be the beginning of this differentiation). However, it is also possible that these theories will continue to behave complexity-wise in a similar fashion, which would both make future analysis easier (in that results for one theory might also be easily adapted to apply to the others) and hint at a unity in explanations within the SMT framework of what makes analogical problem solving computationally “easy.”

In any case, the complexity results of all three processes above bring home what is probably the major take-home messages of this paper with respect to the model of analogical problem solving described in Section 1, namely,

1. all processes invoked by this model (and hence the model itself) are intractable in general, and
2. all of these processes (and hence the model itself) become tractable under plausible constraints, but not in the ways proposed to date in the literature.

Here, (1) establishes that explaining the speed of analogical problem solving under certain (experimental) conditions can be a real challenge for SMT-based models of the underlying processes; whereas (2) establishes that there do exist such explanations but that these have not previously been identified in the literature. The latter observation underscores the utility of performing rigorous computational-level complexity analyses, as we have done in this paper—namely, because such analyses allow definitive statements of which theoretical constraints do and do not yield valid explanations of intractability. Although simulation results obtained by running specific SMT-inspired algorithms can suggest such explanations, one can never rule out that the simulation findings fail to generalize to untested inputs and/or to more efficient algorithmic implementations. Computational-level results as we have derived in this paper, on the other hand, apply to all inputs and the best possible algorithms. Our results thus establish the (im)possibility of tractable computation of modeled sub-processes in a way that no computer simulation can.

Importantly, our claim that the explanations identified by our analyses are “valid” is not to suggest that we claim that these explanations are also “veridical.” They are “valid” only in the sense that they identify a set of constraints that are provably *sufficient* for making the modeled sub-processes (here, EXEMPLAR RETRIEVAL and PROJECTION DERIVATION) tractable. Evidently, our analyses cannot show that one such particular set of constraints is also *necessary* for tractability; in fact, this is contradicted by the fact that we have found multiple such sufficient explanations, and more sufficient explanations may even be found when considering more and different parameters than we have done here. Also, we cannot know which of the sufficient constraints are actually exploited by human problem solvers. To establish veridicality of our theoretically valid explanations, empirical research is needed. One way of providing evidence for or against an explanation of tractability is to systematically manipulate the critical parameters in an experimental design, and test whether performance indeed deteriorates (in terms of speed and/or accuracy) as predicted by that particular explanation. If such a prediction were to be borne out for one of the explanations we have put forth here, this would not only help validate our complexity-theoretic methodology, but also provide corroborating evidence that SMT (and, in particular, our computational-level version of this theory) correctly characterizes the sub-processes underlying human analogical problem solving.

## 6. Conclusion

In this paper we have shown that theories of the sub-processes underlying analogical problem solving, based on SMT, encounter the theoretical obstacle of intractability, *i.e.*, the postulated computations consume an unrealistic amount of time unless the input domain of such theories is properly constrained. Using a complexity-theoretic methodology we



have identified sets of constraints that are provably sufficient for rendering SMT-theories of analogical problem solving tractable. This means that, by enriching SMT-theories with such sets of constraints, the theories can explain the speed of human analogical problem solving under different conditions despite the intractability of the theory for unrestricted domains. Which of the possible explanations put forth in this paper accurately captures how humans succeed in making their analogical problem-solving processes fast enough to be tractable is an interesting open question, which we hope experimental problem-solving researchers find a way to address.

Although we have focused in this paper on analogical problem solving, we emphasize that the applicability of our methodology ranges much wider than that. Many models of so-called domain-general cognition—including problem solving, reasoning, and decision making—typically face the theoretical obstacle of computational intractability, if only because it is nontrivial to characterize domain-generality without falling into the trap of overgeneralizing the input domain beyond the cognitively relevant. Our complexity-theoretic methodology is a general tool for remedying this problem in cognitive theorizing. Based on analyses such as reported in this paper and elsewhere (*e.g.*, Blokpoel et al., 2010; Müller, van Rooij, & Wareham, 2009; van Rooij et al., 2008; van Rooij, Stege, & Kadlec, 2005; Wareham & van Rooij, to appear), we submit that theories of cognitive processes can safely be assumed to be domain-general as long as some minimal input constraints can be assumed that suffice for making such processes tractable, at least in those modeled situations in which people are also fast. We believe this perspective can be seen as an important third alternative to the now standard antagonism between, on the one hand, denying the domain-generality of any cognitive process (*e.g.*, by assuming a massively modular architecture of mind; Carruthers, 2006) and, on the other hand, the stance that domain-general cognition necessarily needs to cut corners in order to be fast (*e.g.*, by using frugal heuristics; Gigerenzer, 2008).

## References

- Arora, S. (1998). The approximability of NP-hard problems. *Proceedings of the Thirtieth ACM Annual Symposium on the Theory of Computing* (pp. 337-348). ACM Press.
- Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., & Protasi, M. (1999). *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer.
- Blanchette, I., & Dunbar, K. (2001). Analogy use in naturalistic settings: The influence of audience, emotions, and goals. *Memory & Cognition*, 29, 730-735.
- Blokpoel, M., Kwisthout, J., van der Weide, T., & van Rooij, I. (2010). How action understanding can be rational, Bayesian, and tractable. *Proceedings of the 32nd Annual Conference of the Cognitive Science Society* (pp. 50-55). Cognitive Science Society.

- Bowdle, B., & Gentner, D. (2005). The career of metaphor. *Psychological Review*, 112(1), 193-216.
- Carbonell, J. G. (1986). Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine Learning: An Artificial Intelligence Approach* (Vol. 2). Morgan Kaufmann.
- Carruthers, P. (2006). The case for massively modular models of mind. In R. J. Stainton (Ed.), *Contemporary Debates in Cognitive Science* (pp. 3-21). Wiley-Blackwell.
- Chater, N., Oaksford, M., Nakisa, R., & Redington, M. (2003). Fast, frugal and rational: How rational norms explain behavior. *Organizational Behavior and Human Decision Processes*, 90, 63-86.
- Chater, N., Tenenbaum, J. B., & Yuille, A. (2006). Probabilistic models of cognition: Conceptual foundations. *Trends in Cognitive Sciences*, 10 (7), 287-201.
- Clement, C. A., & Gentner, D. (1991). Systematicity as a selection constraint in analogical mapping. *Cognitive Science*, 15, 89-132.
- Downey, R., & Fellows, M. (1999). *Parameterized Complexity*. Springer.
- Dunbar, K. (2001). The analogy paradox: Why analogy is so easy in naturalistic settings yet so difficult in the psychological laboratory. In D. Gentner, K. J. Holyoak, & B. N. Kokinov (Eds.), *The Analogical Mind: Perspectives from Cognitive Science* (pp. 313-334). MIT Press.
- Duncker, K. (1945). On problem-solving (L. S. Lees, Trans.). *Psychological Monographs*, 59, Whole No. 270.
- Falkenhainer, B. (1989). *Learning from Physical Analogies: A Study in Analogy and the Explanation Process*. Ph.D. thesis, Department of Psychology, University of Illinois at Urbana-Champaign.
- Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial Intelligence*, 41, 1-63.
- Forbus, K. D., & Gentner, D. (1989). Structural evaluation of analogies: What counts? *Proceedings of the 11th Annual Conference of the Cognitive Science Society* (pp. 341-348). Erlbaum.
- Forbus, K. D., Gentner, D., Everett, J. O., & Wu, M. (1997). Towards a computational model of evaluating and using analogical inferences. *Proceedings of the 19th Annual Conference of the Cognitive Science Society* (pp. 229-234). Erlbaum.
- Forbus, K. D., Gentner, D., & Law, K. (1994). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, 19, 141-205.
- Forbus, K. D., & Hinrichs, T. (2006). Companion cognitive systems: A step towards human-level AI. *AI Magazine*, 27 (2), 83-95.
- Forbus, K. D., & Oblinger, D. (1990). Making SME greedy and pragmatic. *Proceedings of the 12th Annual Conference of the Cognitive Science Society* (pp. 61-68). Erlbaum.
- Fortnow, L. (2009). The status of the P versus NP problem. *Communications of the ACM*, 52 (9), 78-86.

- Garey, M. R., & Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.
- Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.
- Gentner, D. (1989). The mechanisms of analogical learning. In S. Vosniadou & A. Ortony (Eds.), *Similarity and Analogical Reasoning* (pp. 197-241). Cambridge University Press.
- Gentner, D. (2002). Analogical reasoning, psychology of. In *Encyclopedia of Cognitive Science* (pp. 106-112). Nature Publishing Group.
- Gentner, D. (2003). Why we're so smart. In D. Gentner & S. Goldin-Meadow (Eds.), *Language in Mind: Advances in the Study of Language and Thought* (pp. 195-235). MIT Press.
- Gentner, D. (2010). Bootstrapping the mind: Analogical processes and symbol systems. *Cognitive Science*, 34, 752-775.
- Gentner, D., & Colhoun, J. (2010). Analogical processes in human thinking and learning. In B. Glatzeder, V. Goel, & A. von Müller (Eds.), *On Thinking: Vol. 2. Towards a Theory of Thinking* (pp. 35-48). Springer-Verlag.
- Gentner, D., Lowenstein, J., Thompson, L., & Forbus, K. D. (2009). Reviving inert knowledge: Analogical abstraction supports relational retrieval of past events. *Cognitive Science*, 33(6), 1343-1382.
- Gentner, D., & Markman, A. (1997). Structure mapping in analogy and similarity. *American Psychologist*, 52(1), 45-56.
- Gentner, D., Rattermann, M. J., & Forbus, K. D. (1993). The roles of similarity in transfer: Separating retrievability and inferential soundness. *Cognitive Psychology*, 25, 524-575.
- Gentner, D., Rattermann, M. J., Markman, A., & Kotvosky, L. (1995). Two forces in the development of relational similarity. In T. J. Simon & G. S. Halford (Eds.), *Developing Cognitive Competence: New Approaches to Process Modelling* (pp. 263-313). Erlbaum.
- Gick, M., & Holyoak, K. J. (1980). Analogical problem-solving. *Cognitive Psychology*, 12, 306-355.
- Gick, M., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology*, 15, 1-38.
- Gigerenzer, G. (2008). Why heuristics work. *Perspectives on Psychological Science*, 3, 20-29.
- Hamilton, M., Müller, M., van Rooij, I., & Wareham, T. (2007). Approximating solution structure. In E. Demaine, G. Gutin, D. Marx, & U. Stege (Eds.), *Dagstuhl Seminar Proceedings no. 07281: Structure Theory and FPT Algorithmics for Graphs, Digraphs, and Hypergraphs*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI).
- Keane, M. T. (1996). On adaptation in analogy: Tests of pragmatic importance and adaptability in analogical problem solving. *Quarterly Journal of Experimental Psychology*, 49, 1062-1085.

- Klenk, M., & Forbus, K. D. (2009). Analogical model formulation for transfer learning in AP physics. *Artificial Intelligence, 173*(18), 1615-1638.
- Kuehne, S., Forbus, K. D., Gentner, D., & Quinn, B. (2000). SEQL: Category learning as progressive abstraction using structure mapping. *Proceedings of the 22nd Annual Conference of the Cognitive Science Society* (pp. 770-775). Erlbaum.
- Kwisthout, J., Wareham, T., & van Rooij, I. (2010). Letter to the Editor: Bayesian intractability is not an ailment that approximation can cure. *Cognitive Science*, manuscript accepted for publication with minor revisions.
- Love, B. C. (2000). A computational level theory of similarity. *Proceedings of the 22nd Annual Conference of the Cognitive Science Society* (pp. 316-321). Erlbaum.
- Markman, A. (1997). Constraints on analogical inference. *Cognitive Science, 21*(4), 373-418.
- Markman, A., & Gentner, D. (2000). Structure mapping in the comparison process. *American Journal of Psychology, 113*(4), 501-538.
- Marr, D. (1981). *Vision: A Computational Investigation into the Human Representation and Processing Visual Information*. W. H. Freeman.
- Marx, D. (2008). Parameterized complexity and approximation algorithms. *Computer Journal, 51*(1), 60-78.
- Medin, D. L., & Ross, B. H. (1989). The specific character of abstract thought: Categorization, problem-solving, and induction. In R. Sternberg (Ed.), *Advances in the Psychology of Human Intelligence* (Vol. 5, pp. 189-223). Erlbaum.
- Müller, M., van Rooij, I., & Wareham, T. (2009). Similarity as tractable transformation. *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 49-55). Cognitive Science Society.
- Novick, L. R. (1988). Analogical transfer, problem similarity, and expertise. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 14*(3), 510-520.
- Novick, L. R., & Holyoak, K. J. (1991). Mathematical problem solving by analogy. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 17*(3), 398-415.
- Ross, B. H. (1989a). Distinguishing types of superficial similarities: Different effects on the access and use of earlier problems. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 15*(3), 456-468.
- Ross, B. H. (1989b). Reminders in learning and instruction. In S. Vosniadou & A. Ortony (Eds.), *Similarity and Analogical Reasoning* (pp. 438-469). Cambridge University Press.
- Schelhorn, S.-E., Griego, J., & Schmid, U. (2006). Transformational and derivational strategies in analogical problem solving. *Cognitive Processing, 8*, 45-55.
- Spellman, B. A., & Holyoak, K. J. (1996). Pragmatics in analogical mapping. *Cognitive Psychology, 31*, 307-346.

- van Lehn, K., & Jones, R. M. (1991). Learning physics via explanation-based learning of correctness and analogical search control. *Proceedings of the Eighth International Workshop on Machine Learning* (pp. 132-137). Morgan Kaufmann.
- van Rooij, I. (2008). The tractable cognition thesis. *Cognitive Science*, 32, 939-984.
- van Rooij, I., Evans, P., Müller, M., Gedge, J., & Wareham, T. (2008). Identifying sources of intractability in cognitive models: An illustration using analogical structure mapping. *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (pp. 915-920). Cognitive Science Society.
- van Rooij, I., Stege, U., & Kadlec, H. (2005). Sources of complexity in subset choice. *Journal of Mathematical Psychology*, 49(2), 160-187.
- van Rooij, I., & Wareham, T. (2008). Parameterized complexity in cognitive modeling: Foundations, applications and opportunities. *Computer Journal*, 52(3), 385-404.
- van Rooij, I., Wright, C. D., & Wareham, T. (to appear). Intractability and the use of heuristics in psychological explanation. *Synthese*.
- Veale, T., & Keane, M. T. (1997). The competence of sub-optimal theories of structure mapping on hard analogies. *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97)* (Vol. 1, pp. 232-237). Morgan Kaufmann.
- Wareham, T., & van Rooij, I. (to appear). On the computational challenges of analogy-based generalization. *Cognitive Systems Research*.

## Appendix:

In order to prove tractability and intractability results for the theories of relationally-based solved-exemplar retrieval and maximally-relevant projection derivation formulated in Sections 3.1 and 4.1, we must formalize the various entities mentioned in these theories. Hence, in this appendix, we first review the graph-based formalization of analogy derivation given in van Rooij et al. (2008) (Section A.1). This is followed by a brief summary of techniques for establishing polynomial-time and fixed-parameter intractability (Section A.2) and the proofs of the various tractability and intractability results presented in Sections 5, 3.2, and 4.2 (Sections A.3, A.4, and A.5, respectively).

### A.1. Definitions

Predicate-structures are formalized as a restricted type of vertex- and edge-labeled directed acyclic-graph.

**Definition 3** A **concept-graph** is a quadruple  $(G, \lambda_A, \lambda_B, \lambda_P)$  for a directed acyclic-graph  $G = (V, A)$  and functions  $\lambda_A, \lambda_B$ , and  $\lambda_P$  called labelings such that:

1.  $\lambda_A: A \rightarrow \mathcal{N}$ .
2.  $\lambda_B$  is 1:1 and onto and defined on the leaves of  $G$ .
3.  $\lambda_P$  is defined on the internal vertices of  $G$ .
4. If  $v$  is an internal vertex, then  $\lambda_A$  either enumerates the arcs leaving  $v$  or is constantly 0 on this set. In the first case,  $v$  is **ordered**, and in the second **unordered**.
5. For internal vertices  $u, v$  with  $\lambda_P(u) = \lambda_P(v)$ , the following hold:
  - a) Either both  $u$  and  $v$  are ordered or  $u$  and  $v$  are unordered;
  - b)  $u$  and  $v$  both have the same number of children in  $G$ ; and
  - c)  $\{(v', \lambda_A(v, v')) \mid (v, v') \in A\} \neq \{(u', \lambda_A(u, u')) \mid (u, u') \in A\}$ .

A concept-graph is ordered if all of its internal vertices are ordered; otherwise, if it contains at least one unordered predicate, it is unordered. Note that in a concept-graph, internal vertices correspond to predicates or functions, leaves correspond to objects, and the labelings  $\lambda_P$ ,  $\lambda_B$ , and  $\lambda_A$  assign predicate-types, object-names, and predicate argument-order labels, respectively. For later, let us define  $P$  to be the range of  $\lambda_P$ , i.e.,  $P$  is the set of predicate-types in the concept-graph.

Analogies between pairs of concept-graphs are formalized in terms of a restricted type of sub-graph isomorphism. Recall that a graph isomorphism between directed graphs  $G = (V, A)$  and  $G' = (V', A')$  is a 1:1 onto function  $f: V \rightarrow V'$  such that  $(u, v) \in A$  if and only if  $(f(u), f(v)) \in A'$ , and a sub-graph isomorphism between directed graphs  $G$  and  $G'$  is a graph isomorphism  $f$  between a sub-graph of  $G$  and a sub-graph of  $G'$ . Let  $G_f = (V_f, A_f)$  be the directed graph defined by such a sub-graph isomorphism  $f$  (as such,  $G_f$  is the domain of  $f$ , i.e., the sub-graph of  $G$  matched by  $f$ ; in turn, let  $G'$  denote the image of  $f$ , i.e., the sub-graph of  $G'$  matched by  $f$ ).

**Definition 4** Given a pair of concept-graphs  $\mathcal{G} = (G = (V, A), \lambda_A, \lambda_B, \lambda_P)$  and  $\mathcal{G}' = (G' = (V', A'), \lambda_{A'}, \lambda_{B'}, \lambda_{P'})$ , an **analogy-morphism** of  $\mathcal{G}$  and  $\mathcal{G}'$  is a sub-graph isomorphism  $f$  that satisfies the following three conditions:

1. For all  $v \in V_f$ , all children of  $v$  in  $G$  are also in  $V_f$ ;
2. For all  $v \in V_f$ ,  $\lambda_{P'}(f(v)) = \lambda_P(v)$ ; and
3. For all  $(v, w) \in A_f$ ,  $\lambda_{A'}((f(v), f(w))) = \lambda_A((v, w))$ .

Note that the various conditions used by Gentner to define an analogy are encoded in this definition—in particular, the fact that an analogy-morphism is a 1:1 onto function ensures one-to-one correspondence, conditions (1) and (3) ensure parallel connectivity, and condition (2) ensures relational focus.

Systematicity is assessed using the following function on concept-graphs:

**Definition 5** Given a concept-graph  $\mathcal{G} = (G = (V, A), \lambda_a, \lambda_B, \lambda_p)$ , a function  $pval: P \rightarrow \mathcal{N}$  and positive numbers  $trd, lm \in \mathcal{N}$ , the **value** of  $\mathcal{G}$  is  $val(\mathcal{G}) = \sum_{v \in V} val(v)$  where  $val(v) = match(v) + \sum_{(w,v) \in A} trd \times val(w)$  such that  $match(v)$  is  $pval(\lambda_p(v))$  if  $v$  is an internal vertex and  $lm$  if  $v$  is a leaf.

Given the above, the value of an analogy-morphism  $f$  of two concept-graphs is  $val(G_f)$ . Note that the function  $val()$  defined above is essentially that used to evaluate systematicity in SME, where  $pval()$  is a function assigning match-values to individual predicate-types,  $trd$  is the trickle-down factor by which the values of parent-predicates are multiplied when added to compute a child-predicates score, and  $lm$  is the match-value of objects (see Forbus & Gentner, 1989, and Forbus et al., 1994, p. 158 for details).

Given the above, we can formalize all definitions and problems given in Sections 3.1, 4.1, and 5 by substituting concept-graphs for predicate-structures, analogy-morphisms for analogies, and  $\max_{f \in F(x,y)} val(G_f)$  for  $SE(x, y)$ , where  $F(x,y)$  is the set of all possible analogy-morphisms between  $x$  and  $y$ .

### A.2. Proving Intractability

Given some criterion of tractability like polynomial-time or fixed-parameter solvability, we can define the class  $T$  of all input-output mappings that are tractable relative to that criterion. For example,  $T$  could be the class  $P$  of decision problems (see below) solvable in polynomial-time, or  $FPT$ , the class of parameterized problems that are fp-tractable. We can show that a particular input-output mapping is not in  $T$  (and thus that this mapping is intractable) by showing that this mapping is at least as hard as the hardest input-output mapping in any mapping-class  $C$  that properly includes (or is strongly conjectured to properly include)  $T$ . For example,  $C$  could be  $NP$ , the class of decision problems whose candidate solutions can be verified in polynomial time, or a class of parameterized problems in the  $W$ -hierarchy =  $\{W[1], W[2], \dots, W[P], \dots, XP\}$  (see Garey & Johnson, 1979, and Downey & Fellows, 1999 for details).

We will focus here on reducibilities between pairs of **decision problems**, *i.e.*, input-output mappings whose answers are either "Yes" or "No." The two types of reductions used in this paper are as follows.

**Definition 6** Given a pair  $\Pi, \Pi'$  of decision problems,  $\Pi$  **polynomial-time many-one reduces to**  $\Pi'$  if there is a polynomial-time computable function  $f$  mapping instances  $I$  of  $\Pi$  to instances  $f(I)$  of  $\Pi'$  such that the answer to  $I$  is "Yes" if and only if the answer to  $f(I)$  is "Yes."

**Definition 7** Given a pair  $\Pi, \Pi'$  of parameterized decision problems,  $\Pi$  **fp-reduces to**  $\Pi'$  if there is a function  $f$  mapping instances  $I = (x, p)$  of  $\Pi$  to instances  $I' = (x', p')$  of  $\Pi'$  such that (i)  $f$  is computable in  $g(p)|x|^\alpha$  time for some function  $g()$  and constant  $\alpha$ , (ii)  $p' = h(p)$  for some function  $h()$ , and (iii) the answer to  $I$  is “Yes” if and only if the answer to  $I' = f(I)$  is “Yes.”

A reducibility is appropriate for a tractability class  $T$  if whenever  $\Pi$  reduces to  $\Pi'$  and  $\Pi' \in T$  then  $\Pi \in T$ . We say that a problem  $\Pi$  is **C-hard** for a class  $C$  if every problem in  $C$  reduces to  $\Pi$ . A  $C$ -hard problem is essentially as hard as the hardest problem in  $C$ .

Reducibilities become particularly useful in light of the following three easily-provable properties:

1. If  $\Pi$  reduces to  $\Pi'$  and  $\Pi$  is  $C$ -hard then  $\Pi'$  is  $C$ -hard.
2. If  $\Pi$  is  $C$ -hard and  $T \subset C$  then  $\Pi \notin T$ , i.e.,  $\Pi$  is not tractable.
3. If  $\Pi$  is  $C$ -hard and  $T \subseteq C$  then  $\Pi \notin T$  unless  $T = C$ , i.e.,  $\Pi$  is not tractable unless  $T = C$ .

These properties are easily provable for many commonly-used reducibilities, including those given in Definitions 6 and 7 above. The first and third of these properties will be used to show intractability below relative to tractable  $T$ -classes  $P$  and  $FPT$  and enclosing but not provably properly enclosing  $C$ -classes  $NP$ ,  $W[1]$ , and  $XP$ . Note that these intractability results hold relative to the conjectures  $P \neq NP$  and  $FPT \neq W[1]$  which, though not proved, have strong empirical support and are commonly accepted as true within the Computer Science community (see Fortnow, 2009, Garey & Johnson, 1979, and Downey & Fellows, 1999 for details).

In the remainder of this paper, as we will be using reducibilities that operate on decision problems, we will actually be showing results relative to the decision problems associated with the input-output mappings `EXEMPLAR RETRIEVAL` and `PROJECTION DERIVATION` (see Sections A.4 and A.5 below). This is acceptable for such suitably-defined decision problems because if such a problem is intractable, then so is the associated input-output mapping (otherwise, any tractable algorithm computing the input-output mapping could be used to construct a tractable algorithm for solving the decision problem, which would contradict the intractability of the decision problem).

Note also that in subsequent sections, all intractability results state specifically whether they hold relative to concept-graphs that are ordered or unordered. This is done for formal completeness. However, given the focus in the main text on the more general unordered case and the fact that any result relative to ordered concept-graphs implies the same result for unordered graphs (namely, add an extra unordered predicate of type not previously used in the instance to any concept-graph to make the instance unordered), the results cited in the main text do not mention concept-graph order.



### A.3. New *fp*-Tractability Results for Analogy Derivation

These results are derived relative to the following formal input-output mapping associated with ANALOGY DERIVATION as defined in Section 5:

ANALOGY DERIVATION

*Input:* Two concept-graphs  $B$  and  $T$ .

*Output:* An analogy-morphism  $A$  between  $B$  and  $T$  of optimal value.

**Result AD5:** *Unordered and ordered ANALOGY DERIVATION are *fp*-tractable for parameter-set  $\{p\}$ .*

**Proof:** The essential idea underlying this proof is that an analogy-morphism can be constructed in two phases—first, a partial-analogy-morphism  $m$  can be drawn consisting only of matching predicates in the given concept-graphs  $\mathcal{G}$  and  $\mathcal{G}'$ , and second, this partial-analogy-morphism can be augmented to create an analogy-morphism proper by adding an appropriate morphism between the sets of leaf-objects connected to the predicate-sets  $X$  and  $X'$  linked by the partial analogy-morphism. The key to efficiently computing this appropriate morphism is to realize that even though there may be several leaf-objects in  $\mathcal{G}$  with the same predicate-parents in  $X$ , *any* of these can be matched to *any* of the corresponding leaf-objects in  $\mathcal{G}'$  with the same predicate-parents and predicate-argument indices under  $m$  to create a  $m$ -valid analogy-morphism.

Our proof will use the following notations. Given concept-graphs  $\mathcal{G} = (G, \lambda_A, \lambda_P, \lambda_B)$  and  $\mathcal{G}' = (G', \lambda_{A'}, \lambda_{P'}, \lambda_{B'})$ , let  $M(\mathcal{G}, \mathcal{G}')$  be the set of all sub-graph isomorphisms between sub-graphs of  $G$  and  $G'$  that are composed entirely of internal vertices in these graphs. Given  $m \in M(\mathcal{G}, \mathcal{G}')$ , recall that  $G_m$  is the domain of  $m$ , that is, the sub-graph of  $G$  matched by  $m$ , and  $G'_m$  is the range of  $m$ , that is, the sub-graph of  $G'$  matched by  $m$ . Define a **pseudo-concept-graph** as any sub-graph of a concept-graph consisting entirely of internal vertices in the parent graph that satisfies all conditions in the definition of a concept-graph except those relating to leaf-objects. Pseudo-concept-graphs are thus sub-concept-graphs with the leaf-objects stripped off. Note that some (but not all)  $m \in M(\mathcal{G}, \mathcal{G}')$  correspond to morphisms between pseudo-concept-graphs; for such  $m$ , let  $\mathcal{G}_m$  and  $\mathcal{G}'_m$  denote these pseudo-concept-graphs in  $\mathcal{G}$  and  $\mathcal{G}'$ , respectively. Finally, given a pseudo-concept-graph  $\mathcal{P}$  of a concept-graph  $\mathcal{G}$ , let  $Obj(\mathcal{P}, \mathcal{G})$  be the set of all leaf-objects in  $\mathcal{G}$  whose parent-predicates are all in  $\mathcal{P}$ .

Given the above, consider the algorithm AD<sub>p</sub> given in Table 3. Observe that if elements  $x$  and  $y$  of sets  $L$  and  $L'$  are equal, the corresponding leaf-objects  $o_x$  and  $o_y$  in  $\mathcal{G}$  and  $\mathcal{G}'$  are functionally identical with respect to  $m$ , that is, the leaf-objects have the same parent-predicates and serve the same predicate argument-order role for each such parent-predicate. Moreover, if  $L = L'$ , any matching of identical elements in  $L$  and  $L'$  comprises

**Table 3.** The analogy derivation (relative to parameter  $\{p\}$ ) (ADp) Algorithm.

- Procedure ADp( $\mathcal{G}, \mathcal{G}'$ )
1.  $best = \emptyset$
  2. for  $m \in M(\mathcal{G}, \mathcal{G}')$
  3. if  $\mathcal{G}_m$  and  $\mathcal{G}'_m$  are pseudo-concept-graphs
  4.  $L = L' = \emptyset$
  5. for  $v \in Obj(\mathcal{G}_m, \mathcal{G})$
  6.  $par(v) = \{(m(x), AA((x, v)) \mid (x, v) \in A\}$
  7.  $L = L \cup \{par(v)\}$
  8. for  $v' \in Obj(\mathcal{G}'_m, \mathcal{G}')$
  9.  $par(v') = \{(x', \lambda_{A'}((x', v')) \mid (x', v') \in A'\}$
  10.  $L' = L' \cup \{par(v')\}$
  11. if  $L = L'$
  12. Extract from any matching of identical elements of  $L$  and  $L'$   
a 1:1 onto function  $g$  from  $Obj(\mathcal{G}_m, \mathcal{G})$  to  $Obj(\mathcal{G}'_m, \mathcal{G}')$  such that  
 $par(v) = par(h(v))$ .
  13. Augment  $m$  with  $g$  to create an analogy-morphism for  $\mathcal{G}$  and  $\mathcal{G}'$ .
  14. if ( $best = \emptyset$ ) or ( $val(G_f) > val(G_{best})$ )
  15.  $best = f$
  16. return  $best$

a morphism between leaf-objects in  $\mathcal{G}$  and  $\mathcal{G}'$  that augments  $m$  to make a full analogy-morphism relative to the pseudo-concept-graphs  $\mathcal{G}_m$  and  $\mathcal{G}'_m$ .

All operations on lines 1 and 3-16 of the algorithm can be done in time polynomial in the sizes of  $\mathcal{G}$  and  $\mathcal{G}'$ ; moreover, the size of  $M(\mathcal{G}, \mathcal{G}')$  is

$$\begin{aligned}
 \sum_{i=1}^p choose(p, i) \times choose(p, i) \times i! &= \sum_{i=1}^p choose(p, i) \times permute(p, i) \\
 &\leq \sum_{i=1}^p choose(p, i) \times p! \\
 &= p! \times \sum_{i=1}^p choose(p, i) \\
 &= p! \times 2^p \\
 &\leq p^p \times 2^p
 \end{aligned}$$

which is purely a function of  $p$ . Hence, the algorithm runs in time that is exponential purely in  $p$ , completing the proof.  $\square$

**Result AD6:** *Unordered and ordered ANALOGY DERIVATION are fp-tractable for parameter-set  $\{f, t\}$ .*

**Proof:** Follows from the algorithm in the proof of Result AD5 and the observation that  $p \leq f \times t$ .  $\square$

**Result AD7:** *Unordered and ordered ANALOGY DERIVATION are fp-tractable for parameter-sets  $\{|B|, f\}$  and  $\{|T|, f\}$ .*

**Proof:** Follows from the algorithm in Result AD5 and the observations that (i) the maximum number of possible alignable predicates in  $T$  with respect to  $B$  (and hence  $p$ )  $\leq |B| \times f$ , and (ii) the maximum number of possible alignable predicates in  $B$  with respect to  $T$  (and hence  $p$ )  $\leq |T| \times f$ .  $\square$

#### A.4. Results for Exemplar Retrieval

These results are derived relative to the following formal decision problem associated with EXEMPLAR RETRIEVAL as defined in Section 3.1:

EXEMPLAR RETRIEVAL

*Input:* A library  $L$  of concept-graphs, a probe concept-graph  $P$ , and integers  $b$  and  $k$ .

*Question:* If  $L'$  is the largest possible subset of  $L$  of size  $\leq b$  such that each member of  $L'$  has an optimal analogy-morphism with  $P$  of value that is the largest possible over the values of all optimal analogy-morphisms between  $P$  and members of  $L$ , is  $L'$  of size  $\leq k$ ?

Our intractability results will be derived relative to reductions from the following problems:

CLIQUE

*Input:* A graph  $G$  and an integer  $k$ .

*Question:* Does  $G$  contain a CLIQUE with  $k$  vertices, that is, is there a sub-graph of  $G$  that is isomorphic to  $K_k$ , the complete graph on  $k$  vertices?

3-DIMENSIONAL MAPPING

*Input:* A set  $M \subseteq X \times Y \times Z$ , where  $X$ ,  $Y$ , and  $Z$  are disjoint sets having the same number  $q$  of elements.

*Question:* Is there a subset  $M' \subseteq M$  such that  $M'$  is a matching, that is, no two elements of  $M'$  agree in any co-ordinate, and  $|M'| = q$ ?

These reductions will use the following transformation to create concept-graphs:

- Given an unordered graph  $G = (V, E)$ , let  $C(G)$  be the unordered concept-graph consisting of  $|V|$  objects and  $|E|$  unordered binary predicates of type  $p$  in which each object corresponds to a unique vertex  $G$  and each predicate  $a$  corresponds to a unique edge  $e = (u, v)$  in  $E$  such that the arguments of  $a$  are the objects corresponding to the endpoints  $u$  and  $v$  of  $e$  in  $G$ .
- Given a set  $M \subseteq X \cup Y \cup Z$  for disjoint sets  $X$ ,  $Y$ , and  $Z$  with the same number of elements, let  $CMA(M)$  be the ordered concept-graph consisting of  $|X| + |Z|$  objects and  $|M|$  ordered binary predicates of distinct predicate-type in which each object corresponds to a unique element in  $X \cup Z$  and each predicate  $a$  corresponds to a unique element  $m = (x, y, z)$  of  $M$  such that  $a$  has unique predicate-type  $p_{xyz}$  and the first and second arguments of  $a$  are the objects corresponding to  $x$  and  $z$ , respectively.
- Given a set  $M \subseteq X \cup Y \cup Z$  for disjoint sets  $X$ ,  $Y$ , and  $Z$  with the same number of elements, let  $CMB(M)$  be the ordered concept-graph consisting of  $|Y| + (|X| \times |Y|)$  objects and  $|M|$  ordered binary predicates of distinct predicate-type in which each object corresponds to a unique element of  $Y \cup \{v_{xy} \mid \text{there exists an element } (x, y, z) \in M\}$  and each predicate  $a$  corresponds to a unique element  $m = (x, y, z)$  of  $M$  such that  $a$  has unique predicate-type  $p_{xyz}$  and the first and second arguments of  $a$  are the object corresponding to  $y$  and  $v_{xy}$ , respectively.

Note that  $C(G)$  was originally defined in the Supplementary Materials of van Rooij et al. (2008) and  $CMA(M)$  and  $CMB(M)$  were used but not defined as above in Veale and Keane (1997).

**Lemma 1:** *CLIQUE polynomial-time many-one reduces to Unordered EXEMPLAR RETRIEVAL such that in the constructed instance of Unordered EXEMPLAR RETRIEVAL,  $|B|$  is a function of  $k$  from the given instance of CLIQUE,  $h = 1$ ,  $a = 2$ ,  $t = 1$ ,  $|L| = 2$ , and  $b = 2$ .*

**Proof:** Given an instance  $I = \langle G, k \rangle$  of CLIQUE, construct the following instance  $I' = \langle L, P, b, k' \rangle$  of Unordered EXEMPLAR RETRIEVAL: let  $L = \{C(K_k), C(K_{k-1})\}$ ,  $P = C(G)$ ,  $b = 2$ , and  $k' = 1$ . Note that  $I'$  can be constructed in time polynomial in the size of  $I$ ; moreover, in  $I'$ ,  $|B| = k + k(k-1)/2$ .

To prove that this construction is a reduction, we must show that the answer to the given instance of CLIQUE is “Yes” if and only if the answer to the constructed instance of EXEMPLAR RETRIEVAL is “Yes.” We will do this by proving both directions of this implication separately: if the answer to the given instance of CLIQUE is “Yes,” there is a  $k$ -CLIQUE in  $G$ ; as  $L'$  can only contain concept-graphs that yield optimal analogies to  $P$ ,  $L'$  will contain one element, that is,  $C(K_k)$ , and the answer to the constructed instance of EXEMPLAR RETRIEVAL will be “Yes.” Conversely, if the answer to the constructed instance of EXEMPLAR RETRIEVAL is “Yes,”

$|L'| = 1$ , and the sole member of  $L'$  is  $C(K_k)$ . This can be proved by contradiction: if  $C(K_{k-1})$  was the sole member, this would mean that the largest possible analogy of a member of  $L$  and  $P$  is a (possibly proper) sub-graph of  $C(K_{k-1})$ ; however, as any such sub-graph also exists within  $C(K_k)$ ,  $C(K_k)$  would also yield an optimal analogy to  $P$ , which contradicts the definition of  $L'$  as the largest possible subset of  $L$  having optimal analogies with  $P$ . As the sole member of  $L'$  is  $C(K_k)$ , this implies that there is a  $k$ -CLIQUE in the graph  $G$ , and that the answer to the given instance of CLIQUE is "Yes," completing the proof.  $\square$

**Lemma 2:** CLIQUE *polynomial-time many-one reduces to Unordered EXEMPLAR RETRIEVAL such that in the constructed instance of Unordered EXEMPLAR RETRIEVAL,  $|T|$  is a function of  $k$  from the given instance of CLIQUE,  $h = 1$ ,  $a = 2$ ,  $t = 1$ ,  $|L| = 2$ , and  $b = 2$ .*

**Proof:** Given an instance  $I = \langle G, k \rangle$  of CLIQUE, construct the following instance  $I' = \langle L, P, b, k' \rangle$  of Unordered EXEMPLAR RETRIEVAL: let  $L = \{C(G), C(K_{k-1})\}$ ,  $P = C(K_k)$ ,  $b = 2$ , and  $k' = 1$ . The remainder of the proof is analogous to that for Lemma 1.  $\square$

**Lemma 3:** 3-DIMENSIONAL MATCHING *polynomial-time many-one reduces to Unordered and Ordered EXEMPLAR RETRIEVAL such that in the constructed instance of EXEMPLAR RETRIEVAL,  $h = 1$ ,  $a = 2$ ,  $f = 1$ , and  $b = 2$ .*

**Proof:** Consider first the case of Ordered EXEMPLAR RETRIEVAL. Given an instance  $I = \langle M, X, Y, Z \rangle$  of 3-DIMENSIONAL MATCHING, construct the following instance  $I' = \langle L, P, b, k \rangle$  of Ordered EXEMPLAR RETRIEVAL: let  $L = \{CMA(M)\} \cup \{CMA(M(w)) \mid w \in X \cup Y \cup Z\}$  where  $M(w)$  is the subset of  $M$  in which all triples including  $w$  have been removed,  $P = CMB(M)$ ,  $b = 4$ , and  $k = 1$ . Note that  $I'$  can be constructed in time polynomial in the size of  $I$ .

To prove that this construction is a reduction, we must show that the answer to the given instance of 3-DIMENSIONAL MATCHING is "Yes" if and only if the answer to the constructed instance of EXEMPLAR RETRIEVAL is "Yes." We will do this by proving both directions of this implication separately: If the answer to the given instance of 3-DIMENSIONAL MATCHING is "Yes," then by the proof given in Veale and Keane (1997, Section 3), there is a matching  $M'$  of size  $q$ . Note that  $M'$  induces an optimal analogy between  $P$  and  $CMA(M)$ , and cannot induce an optimal analogy with any other member of  $L$  as each such member is missing at least one element from  $X \cup Y \cup Z$  and thus can only have a matching with  $M$  of size  $q' < q$  (which induces an analogy of value less than that for  $M'$ ). As  $|L'| = 1$ , the answer to the constructed instance of EXEMPLAR RETRIEVAL is "Yes." If the answer to the constructed instance of EXEMPLAR RETRIEVAL is "Yes,"  $|L'| = 1$ . As any matching  $M'$  of size smaller than  $q$  must leave out the elements in at least one triple in  $M$ , at least four possible members of  $L$  would induce optimal analogies (namely,  $CMA(M)$  and the three  $CMA(M(w))$  such that  $w$  is in the excised triple). Hence, if  $|L'| = 1$ , the optimal analogy must be between  $CMA(M)$  and  $P$ , which corresponds to a matching of size  $q$  in  $M$ . Hence, the answer to the given instance of 3-DIMENSIONAL MATCHING is also "Yes," completing the proof.

As for Unordered EXEMPLAR RETRIEVAL, modify the constructed instance of EXEMPLAR RETRIEVAL such that the two members of  $L$  and  $P$  each contain an additional unordered predicate connecting an arbitrary pair of objects in those respective structures that is of different predicate-type in each structure. These modified predicate-structures are now unordered; moreover, none of these extra predicates can match themselves or any other predicate in  $L$  and  $P$ . Hence, the proof given above still applies in this case.  $\square$

**Lemma 4:** *Unordered EXEMPLAR RETRIEVAL is NP-hard when  $h = 1$ ,  $a = 2$ ,  $t = 1$ ,  $|L| = 2$ , and  $b = 2$ .*

**Proof:** Follows from the NP-hardness of CLIQUE (Garey & Johnson, 1979, Problem GT19) and either of the reductions in Lemmas 1 and 2.  $\square$

**Lemma 5:** *Unordered and Ordered EXEMPLAR RETRIEVAL are NP-hard when  $h = 1$ ,  $a = 2$ ,  $f = 1$ , and  $b = 2$ .*

**Proof:** Follows from the NP-hardness of 3-DIMENSIONAL MATCHING (Garey & Johnson, 1979, Problem SP1) and the reduction in Lemma 3.  $\square$

**Result ER1:** *Unordered and Ordered EXEMPLAR RETRIEVAL are polynomial-time intractable.*

**Proof:** Follows from Lemmas 4 and 5.  $\square$

**Result ER2:** *Unordered and Ordered EXEMPLAR RETRIEVAL are fp-intractable for parameter-set  $\{|B|, h, a, t, |L|, b\}$  when  $h = 1$ ,  $a = 2$ ,  $t = 1$ , and  $|L| = b = 2$ .*

**Proof:** The fp-intractability of Unordered EXEMPLAR RETRIEVAL follows from the  $W[1]$ -hardness of  $\langle k \rangle$ -CLIQUE (Downey & Fellows, 1999) and the reduction in Lemma 1. As  $|B|$  is a function of parameter  $k$  in the given instance of  $\langle k \rangle$ -CLIQUE in this reduction, the fp-intractability of Ordered EXEMPLAR RETRIEVAL follows by an argument similar to that given in the proof of Claim 6 in the Supplementary Materials of van Rooij et al. (2008).  $\square$

**Result ER3:** *Unordered and Ordered EXEMPLAR RETRIEVAL are fp-intractable for parameter-set  $\{|T|, h, a, t, |L|, b\}$  when  $h = 1$ ,  $a = 2$ ,  $t = 1$ , and  $|L| = b = 2$ .*

**Proof:** The fp-intractability of Unordered EXEMPLAR RETRIEVAL follows from the  $W[1]$ -hardness of  $\langle k \rangle$ -CLIQUE (Downey & Fellows, 1999) and the reduction in Lemma 2. As  $|T|$  is a function of parameter  $k$  in the given instance of  $\langle k \rangle$ -CLIQUE in this reduction, the fp-intractability of Ordered EXEMPLAR RETRIEVAL follows by an argument similar to that given in the proof of Claim 6 in the Supplementary Materials of van Rooij et al. (2008).  $\square$

**Result ER4:** *Ordered EXEMPLAR RETRIEVAL is fp-intractable for parameter-set  $\{h, a, f, b\}$  when  $h = 1$ ,  $a = 2$ ,  $f = 1$ , and  $b = 2$ .*

**Proof:** Follows from Lemma 5.  $\square$

**Lemma 6:** *If Unordered and Ordered  $\langle X \rangle$ -ANALOGY DERIVATION is fp-tractable for some parameter  $X$ , then Unordered and Ordered  $\langle X \rangle$ -EXEMPLAR RETRIEVAL is also fp-tractable.*

**Proof:** Follows from the observation that any such algorithm for ANALOGY DERIVATION can be used to compute the optimal analogies of  $P$  with each member of  $L$ , from which set  $L'$  could be constructed accordingly. □

**Result ER5:** *Unordered and Ordered EXEMPLAR RETRIEVAL are fp-tractable for parameter-set  $\{o\}$ .*

**Proof:** Follows from the fp-tractable algorithm for Unordered and Ordered  $\langle o \rangle$ -ANALOGY DERIVATION given in van Rooij et al. (2008, Claim 5) and Lemma 6. □

**Results ER6:** *Unordered and Ordered EXEMPLAR RETRIEVAL are fp-tractable for parameter-set  $\{p\}$ .*

**Proof:** Follows from the algorithm in the proof of Result AD5 and Lemma 6. □

**Result ER7:** *Unordered and Ordered EXEMPLAR RETRIEVAL are fp-tractable for parameter-set  $\{f, t\}$ .*

**Proof:** Follows from the algorithm in the proof of Result AD6 and Lemma 6. □

**Result ER8:** *Unordered and Ordered EXEMPLAR RETRIEVAL are fp-tractable for the parameter-sets  $\{|B|, f\}$  and  $\{|T|, f\}$ .*

**Proof:** Follows from the algorithms in the proof of Result AD7 and Lemma 6. □

### **A.5. Results for Projection Derivation**

These results are derived relative to the following formal decision problem associated with PROJECTION DERIVATION as defined in Section 4.2:

PROJECTION DERIVATION

*Input:* Two predicate-structures  $B$  and  $T$ , a subset  $S$  of the predicates in  $B$ , and integers  $k$  and  $d$ .

*Question:* Is there a maximal analogy  $A$  between  $B$  and  $T$  and a projection  $P$  relative to  $A$  and  $B$  such that  $A$  and  $P$  include at least  $k$  members of  $S$  and each candidate inference has distance at most  $d$  from the base portion of the analogy?

Once again, we will derive hardness results by reductions from the problems CLIQUE and 3-DIMENSIONAL MATCHING defined in Section A.4. In addition to concept-graph creation functions  $C(G)$ ,  $CMA(M)$ , and  $CMB(M)$  defined in Section A.4, these reductions will use the following such functions:

- Given an unordered graph  $G = (V, E)$ , let  $CP(G)$  be the concept-graph based on  $C(G)$  in which each edge-predicate in  $C(G)$  modeling edge  $e \in E$  has an associated higher-level binary predicate of a new type  $p'$  that takes as its two arguments the edge-predicate and an  $e$ -specific object  $v_e$  that is the argument of no other predicate in  $CP(G)$ . Note  $CP(G)$  adds  $|E|$  new predicates and  $|E|$  new leaf-objects to  $C(G)$ .
- Given a set  $M \subseteq X \cup Y \cup Z$  for disjoint sets  $X$ ,  $Y$ , and  $Z$  with the same number of elements, let  $CMA(M)$  be the concept-graph based on  $CA(M)$  in which each triple-predicate in  $CA(M)$  modeling triple  $m \in M$  has an associated higher-level binary predicate of distinct type that takes as its two arguments the triple-predicate and an  $m$ -specific object  $v_m$  that is the argument of no other predicate in  $CMA(M)$ . Note that  $CMA(M)$  adds  $|M|$  new predicates and  $|M|$  new leaf-objects to  $CA(M)$ .

**Lemma 7:** *CLIQUE polynomial-time many-one reduces to Unordered PROJECTION DERIVATION such that in the constructed instance of PROJECTION DERIVATION,  $|B|$  and  $|S|$  are functions of  $k$  from the given instance of CLIQUE,  $h = a = t = 2$ , and  $d = 1$ .*

**Proof:** Given an instance  $I = \langle G, k \rangle$  of CLIQUE, construct the following instance  $I' = \langle B, T, S, k', d \rangle$  of Unordered PROJECTION DERIVATION: let  $B = CP(K_k)$ ,  $T = C(G)$ ,  $S$  be the set of new predicates added to  $C(K_k)$  in creating  $CP(K_k)$ ,  $k' = |S|$ , and  $d = 1$ . Note that  $I$  can be constructed in time polynomial in the size of  $I$ ; moreover, in  $I'$ ,  $|S| = k(k-1)/2$  and  $|B| = k + 3k(k-1)/2$ .

To prove that this construction is a reduction, we must show that the answer to the given instance of CLIQUE is “Yes” if and only if the answer to the constructed instance of Unordered PROJECTION DERIVATION is “Yes.” We will do this by proving both directions of this implication separately: if the answer to the given instance of CLIQUE is “Yes,” there is a sub-graph of  $T$  that forms an analogy with  $C(K_k)$  (and hence the  $C(K_k)$ -portion of  $CP(K_k)$ ) such that all of the higher-level edge-specific predicates in  $CP(K_k)$  and their associated edge-specific leaf-objects can be projected from  $B$  to  $T$ . As every inference in this projection has distance 1 from the base analogy, the answer to the constructed instance of PROJECTION DERIVATION is “Yes.” If the answer to the given instance of PROJECTION DERIVATION is “Yes,” all predicates in  $S$  must have been projected relative to the analogy between  $B$  and  $T$ ; however, as each predicate in  $S$  is associated with a single edge-predicate in  $B$ , this can only happen if the analogy between  $B$  and  $T$  includes all edge-predicates in  $B$ , that is, there is a sub-graph of  $T$  that is isomorphic to  $C(K_k)$ . As this can only happen if there is a sub-graph of  $G$  that is isomorphic to  $K_k$ , the answer to the given instance of CLIQUE is “Yes,” completing the proof.  $\square$



**Lemma 8:** CLIQUE *polynomial-time many-one reduces to Unordered* PROJECTION DERIVATION *such that in the constructed instance of* PROJECTION DERIVATION,  $|T|$  *is a function of*  $k$  *in the given instance of* CLIQUE,  $h = a = t = 2$ , and  $d = 1$ .

**Proof:** Given an instance  $I = \langle G, k \rangle$  of CLIQUE, construct the following instance  $I' = \langle B, T, S, k', d \rangle$  of Unordered PROJECTION DERIVATION: let  $B = CP(G)$ ,  $T = C(K_k)$ ,  $S$  be the set of new predicates added to  $C(G)$  in creating  $CP(G)$ ,  $k' = k(k-1)/2$ , and  $d = 1$ . Note that  $I'$  can be constructed in time polynomial in the size of  $I$ ; moreover, in  $I'$ ,  $|T| = k + 3k(k-1)/2$ . The remainder of the proof is analogous for that of Lemma 7.  $\square$

**Lemma 9:** 3-DIMENSIONAL MATCHING *polynomial-time many-one reduces to Ordered* PROJECTION DERIVATION *such that in the constructed instance of* PROJECTION DERIVATION,  $h = a = 2$  and  $f = d = 1$ .

**Proof:** Consider first the case of Ordered PROJECTION DERIVATION. Given an instance  $I = \langle M, X, Y, Z \rangle$  of 3-dimensional mapping, construct the following instance  $I' = \langle B, T, S, k', d \rangle$  of PROJECTION DERIVATION: let  $B = CMAP(M)$ ,  $T = CMB(M)$ ,  $S$  be the set of new predicates added to  $CMA(M)$  in creating  $CMAP(M)$ ,  $k' = q$ , and  $d = 1$ . Note that  $I'$  can be constructed in time polynomial in the size of  $I$ .

To prove that this construction is a reduction, we must show that the answer to the given instance of 3-dimensional mapping is "Yes" if and only if the answer to the constructed instance of Ordered PROJECTION DERIVATION is "Yes." We will do this by proving both directions of this implication separately: if the answer to the given instance of 3-dimensional mapping is "Yes," there is a matching  $M'$  of size  $q$  and a corresponding optimal analogy of  $B = CMAP(M)$  with  $T = CMB(M)$  (or rather, of the  $CMA(M)$ -portion of  $CMAP(M)$  with  $CMB(M)$ ) such that  $q$  of the higher-level triple-specific predicates in  $CMAP(M)$  and their associated edge-specific leaf-objects can be projected from  $B$  to  $T$ . As every candidate inference in this projection has distance 1 from the base analogy, the answer to the constructed instance of PROJECTION DERIVATION is "Yes." If the answer to the given instance of PROJECTION DERIVATION is "Yes,"  $q$  predicates in  $S$  must have been projected relative to the analogy between  $B$  and  $T$ ; however, as each predicate in  $S$  is associated with a single triple-predicate in  $B$ , this can only happen if the analogy between  $B$  and  $T$  includes  $q$  triple-predicates in  $B$ , that is, there is a matching of  $M$  that contains  $q$  tuples. Hence, the answer to the given instance of 3-dimensional mapping is "Yes," completing the proof.

As for Unordered PROJECTION DERIVATION, modify the constructed instance of PROJECTION DERIVATION such that  $B$  and  $T$  each contain an additional unordered predicate connecting an arbitrary pair of objects in those structures that is of different predicate-type in each structure. These modified predicate-structures are now unordered; moreover, none of these extra predicates can match themselves or any other predicate in  $B$  and  $T$ . Hence, the proof given above still applies in this case.  $\square$

**Lemma 10:** *Unordered* PROJECTION DERIVATION *is NP-hard when*  $h = a = t = 2$  *and*  $d = 1$ .

**Proof:** Follows from the NP-hardness of CLIQUE (Garey & Johnson, 1979, Problem GT19) and either of the reductions in Lemmas 7 and 8.  $\square$

**Lemma 11:** *Unordered and Ordered* PROJECTION DERIVATION *are NP-hard when*  $h = a = 2$  *and*  $f = d = 1$ .

**Proof:** Follows from the NP-hardness of 3-dimensional mapping (Garey & Johnson, 1979, Problem SP1) and the reduction in Lemma 9.  $\square$

**Result PD1:** *Unordered and Ordered* PROJECTION DERIVATION *are polynomial-time intractable.*

**Proof:** Follows from Lemmas 10 and 11.  $\square$

**Result PD2:** *Unordered and Ordered* PROJECTION DERIVATION *are fp-intractable for parameter-set*  $\{|B|, h, a, t, |S|, k, d\}$  *when*  $h = a = t = 2$  *and*  $d = 1$ .

**Proof:** The fp-intractability of Unordered PROJECTION DERIVATION follows from the  $W[1]$ -hardness of  $\langle k \rangle$ -CLIQUE (Downey & Fellows, 1999) and the reduction in Lemma 7. As  $|B|$  is a function of parameter  $k$  in the given instance of  $\langle k \rangle$ -CLIQUE in this reduction, the fp-intractability of Ordered PROJECTION DERIVATION follows by an argument similar to that given in the proof of Claim 6 in the Supplementary Materials of van Rooij et al. (2008).  $\square$

**Result PD3:** *Unordered and Ordered* PROJECTION DERIVATION *are fp-intractable for parameter-set*  $\{|T|, h, a, t, k, d\}$  *when*  $h = a = t = 2$  *and*  $d = 1$ .

**Proof:** The fp-intractability of Unordered PROJECTION DERIVATION follows from the  $W[1]$ -hardness of  $\langle k \rangle$ -CLIQUE (Downey & Fellows, 1999) and the reduction in Lemma 8. As  $|T|$  is a function of parameter  $k$  in the given instance of  $\langle k \rangle$ -CLIQUE in this reduction, the fp-intractability of Ordered PROJECTION DERIVATION follows by an argument similar to that given in the proof of Claim 6 in the Supplementary Materials of van Rooij et al. (2008).  $\square$

**Result PD4:** *Ordered* PROJECTION DERIVATION *is fp-intractable for parameter-set*  $\{h, a, f, b\}$  *when*  $h = a = 2$ , *and*  $f = d = 1$ .

**Proof:** Follows from Lemma 11.  $\square$

**Result PD5:** *Unordered and Ordered* PROJECTION DERIVATION *are fp-tractable for parameter-set*  $\{o, |S|\}$ .

**Proof:** The fp-tractable algorithm for  $\langle o \rangle$ -ANALOGY DERIVATION in van Rooij et al. (2008, Claim 5) actually computes the maximal analogies for each possible bijection between the objects of  $B$  and the objects of  $T$ . To evaluate the projection-possibilities for each such maximal analogy  $A$ , for each subset  $x$  of  $S$ , determine if each predicate in  $x$  is a valid candidate projection relative to  $A$  and, moreover, if the projection formed by the union of

these candidate inferences satisfies the conditions laid out in PROJECTION DERIVATION. As the maximal analogies can be computed in time exponential purely in  $o$  and the computation of each possible projection relative to the subsets of  $S$  for each maximal analogy can be computed in time exponential purely in  $|S|$ , the algorithm as a whole runs in FPT time relative to  $o$  and  $|S|$ , completing the proof.  $\square$

**Result PD6:** *Unordered and Ordered PROJECTION DERIVATION are fp-tractable for parameter-set  $\{p\}$ .*

**Proof:** The algorithm in the proof of Result AD5 actually computes all maximal analogies for  $B$  and  $T$ . To evaluate the projection-possibilities for each such maximal analogy  $A$ , use the procedure described in the proof of Result PD5. As  $|S| \leq p$ , the algorithm as a whole runs in time exponential purely in  $p$ , completing the proof.  $\square$

**Result PD7:** *Unordered and Ordered PROJECTION DERIVATION are fp-tractable for parameter-set  $\{f, t\}$ .*

**Proof:** Follows from the algorithm in the proof of Result PD6 and the observation that  $p \leq f \times t$ .  $\square$

**Result PD8:** *Unordered and Ordered PROJECTION DERIVATION are fp-tractable for parameter-sets  $\{|B|, f\}$  and  $\{|T|, f\}$ .*

**Proof:** Follows from the algorithm in Result PD6 and the observations that (i) the maximum number of possible alignable predicates in  $T$  with respect to  $B$  (and hence  $p$ )  $\leq |B| \times f$ , and (ii) the maximum number of possible alignable predicates in  $B$  with respect to  $T$  (and hence  $p$ )  $\leq |T| \times f$ .  $\square$

## Acknowledgments

The authors are grateful to two anonymous reviewers for their feedback on an earlier version of this manuscript. Two of the authors (lvR and TW) would like to thank Robert Goldstone and Zygmunt Pizlo for the invitations to attend the Workshop on New Perspectives in Problem Solving at Purdue University on November 8-9, 2008, where TW presented a very early version of the research reported in this paper. Part of the support for this research was provided by the Dean of Science (Memorial University of Newfoundland) and NSERC Discovery Grant 228104 (TW).

Paper submitted: August 23, 2010

Paper accepted: December 10, 2010